Network QoS Needs of Advanced Internet Applications

A Survey

2002

Internet2 QoS Working Group

A Survey of Network QoS Needs of Advanced Internet Applications
— Working Document —

Dimitrios Miras

Computer Science Department
University College London
Gower St., London WC1E 6BT, UK
E-mail: d.miras@cs.ucl.ac.uk

Advisory Committee

Dr. Amela Sadagic Ben Teitelbaum
Internet2
Advanced Network and Services
{amela,ben}@advanced.org

Dr. Jason Leigh
Electronic Visualization Laboratory
University of Illinois at Chicago
spiff@evl.uic.edu

Prof. Magda El Zarki Haining Liu
Information and Computer Science
University of California, Irvine
elzarki@uci.edu haining@ics.uci.edu

November 2002

Abstract

During the last few years the Internet has grown tremendously and has penetrated all aspects of everyday life. Starting off as a purely academic research network, the Internet is now extensively used for education, for entertainment, and as a very promising and dynamic marketplace, and is envisioned as evolving into a vehicle of true collaboration and a multi-purpose working environment. Although the Internet is based on a best-effort service model, the simplicity of its packet-switched design and the flexibility of its underlying packet forwarding regime (IP) accommodate millions of users while offering acceptable performance. At the same time, exciting new applications and networked services have emerged, putting greater demands on the network. In order to offer a better-than-best-effort Internet, new service models that offer applications performance guarantees have been proposed. While several of these proposals are in place, and many QoS-enabled networks are operating, there is still a lack of comprehension about the precise requirements new applications have in order to function with high or acceptable levels of quality. Furthermore, what is required is an understanding of how network-level QoS reflects on actual application utility and usability.

This document tries to fill this gap by presenting an extensive survey of applications' QoS needs. It identifies applications that cannot be accommodated by today's best-effort Internet service model, and reviews the nature of these applications as far as their behaviour with respect to the network is concerned. It presents guidelines and recommendations on what levels of network performance are needed for applications to operate with high quality, or within ranges of acceptable quality. In tandem with this, the document highlights the central role of applications and application developers in getting the expected performance from network services. The document argues that the network cannot guarantee good performance unless it is assisted by well-designed applications that can employ suitable adaptation mechanisms to tailor their behaviour to whatever network conditions or service model is present. The document also reviews tools and experimental procedures that have been recently proposed to quantify how different levels of resource guarantees map to application-level quality. This will allow network engineers, application developers and other interested parties to design, deploy and parameterise networks and applications that offer increased user utility and achieve efficient utilisation of network resources.

In its present form, the document is primarily focused on audio and video applications. It presents a detailed analysis of the end-to-end performance requirements of applications like audio-video conferencing, voice over IP, and streaming of high quality audio and video, and gives an overview of the adaptation choices available to these applications so that they can operate within a wider range of network conditions.

Contents

Contents ii

List of Figures iii

List of Tables vii

Glossary ix

1 Introduction 1

1.1 What are the advanced applications? . . . 1

1.2 What is quality of service? . . . 2

1.3 The need to classify applications' requirements . . . 3

2 Taxonomy of advanced applications 5

2.1 From application characteristics to application requirements . . . 5

2.1.1 Application task-centric classification . . . 5

2.1.2 User characteristics . . . 8

2.1.3 Elastic, tolerant, and adaptive applications . . . 10

2.2 A taxonomy based on type and interdependencies between media . . . 11

2.3 Types of generic applications . . . 11

2.4 Classes of higher level applications . . . 12

2.4.1 Auditory applications . . . 12

2.4.1.1 Interactive . . . 12

2.4.1.2 Non-interactive or loosely interactive . . . 13

2.4.2 Video-based applications . . . 13

2.4.2.1 Interactive . . . 13

2.4.2.2 Non-interactive . . . 16

2.4.3 Distributed Virtual Environments (DVEs) . . . 17

2.4.4 Tele-immersion . . . 17

2.4.5 Remote control of instruments . . . 18

2.4.6 Grid computing . . . 19
2.5 Example applications and projects . . . 20
2.5.1 Video-based applications . . . 20
2.5.1.1 H.323-based videoconferencing . . . 20
2.5.1.2 Music video recording . . . 21
2.5.2 Tele-immersion and data visualisation . . . 22
2.5.3 Remote control of scientific instruments . . . 22
2.5.4 Data Grid projects . . . 23

3 Behaviour and QoS requirements of audio-visual applications 25
3.1 Introduction . . . 25
3.1.1 What is application quality? . . . 25
3.1.2 Network QoS parameters . . . 26
3.1.3 Application QoS metrics . . . 27
3.2 Quality requirements of interactive voice . . . 29
3.2.1 Effect of delay . . . 29
3.2.2 Effect of jitter . . . 30
3.2.3 Effect of packet loss . . . 30
3.2.4 Additional sources of information on interactive voice applications and VoIP . . . 32
3.3 Quality requirements of audio transmission . . . 33
3.4 QoS requirements of digital video transmission . . . 33
3.4.1 Interactive video . . . 33
3.4.2 Video streaming . . . 35
3.4.3 Effect of network transmission on digital video quality . . . 36
3.4.3.1 Transmission bit-rate . . . 37
3.4.3.2 End-to-end latency and delay variation . . . 38
3.4.3.3 Packet Loss . . . 38
3.4.3.4 Interactions between the media in video-based services . . . 40
3.5 An application-network cooperative approach to application QoS . . . 43
3.5.1 A review of common adaptation techniques for audio and video . . . 46
3.5.1.1 Rate adaptation . . . 46
3.5.1.2 Adaptation to delay and delay variance . . . 47
3.5.1.3 Adaptation and resilience to packet loss . . . 48

4 Measuring application quality: Tools and procedures 51
4.1 Measuring the quality of video . . . 52
4.1.1 Subjective video assessment . . . 52
4.1.1.1 Procedures for subjective quality evaluation . . . 53

4.1.2 Objective metrics of video quality . . . 54
4.1.2.1 Impairments of digital video . . . 55
4.1.2.2 Metrics based on human vision models . . . 56
4.1.2.3 Metrics based on measuring features of perceptual distortions . . . 56
4.1.3 Standardisation efforts . . . 57
4.1.3.1 Video Quality Experts Group . . . 57
4.1.3.2 ITU Study Group 9 . . . 59
4.1.4 Weaknesses of video quality assessment techniques . . . 59
4.2 Measuring the quality of Internet audio . . . 59
4.2.1 Mean Opinion Scores . . . 60
4.2.2 Objective methods of speech quality assessment . . . 60
4.2.2.1 PSQM . . . 61
4.2.2.2 Perceptual Speech Quality Measurement Plus (PSQM+) . . . 61
4.2.2.3 MNB . . . 62
4.2.2.4 PAMS . . . 62
4.2.2.5 PESQ . . . 63
4.2.2.6 The E-model . . . 63

5 Conclusion **65**

Bibliography **67**

List of Figures

2.1 A taxonomy based on the task of the application ***(incomplete...)*** 9

3.1 Effect of one-way delay and packet loss on voice distortion for various voice codecs. Perceived distortion measured using R-ratings, as $100 - R$. (Source: [85]) 31

3.2 Throughput and interactivity requirements for common video services. On the right side of the plot are delay-sensitive interactive video services that may accept slightly higher packet loss rates. On the left side are services that can withstand larger latencies but are less tolerant to packet loss (teledata). Colour of higher intensity illustrates higher requirements for throughput. 34

3.3 Mapping of user-centric requirements for one-way delay and packet loss for audio and video streams. The lower parts of the surfaces depict a user-centric expectation of these performance parameters. The upper parts show how tolerance of delay and loss can be increased, while maintaining user-acceptable quality, by employing packet-loss protection techniques ***(incomplete...).*** 37

3.4 Video quality versus encoding bit rate for three 150-frame-long sequences (*akiyo*, *news* and *rugby*; resolution: 352x288) and two different codecs, H.263 (left) and MPEG-1 (right). Because the *rugby* sequence contains more motion and spatial detail, its quality for equivalent bit-rates is lower in comparison to the other two sequences. 38

3.5 Video quality versus encoding bit-rate under packet loss conditions (constant PLR) for an MPEG-encoded sequence. We observe that the quality is increasing up to a certain bit-rate but then drops as the effect of lost packets increases with higher bit-rates. 40

3.6 Detection of errors in lip-synchronisation for different values of skew (time-difference between audio and video). Also shown in different shading are areas related to the detection of synchronisation errors. (Source: [110], ©IEEE). 41

3.7 Level of annoyance of synchronisation errors for various values of skew. (Source: [110], © IEEE.) 42

3.8 Acceptable region of end-to-end delay for the audio and video parts of a videoconferencing task. Also shown are the areas in which the difference between the audio and video delay is noticeable (loss of audio-video synchronisation). 43

4.1 Video quality assessment scale used in subjective MOS tests 53

4.2 Mapping of R-rating values to MOS, speech transmission quality, and user satisfaction (Note that R values below 50 are not recommended for a speech transmission system). 64

List of Tables

3.1 Delay guidelines for VoIP 29
3.2 Jitter guidelines for VoIP 29
3.3 Effect of packet loss on voice quality 31
3.4 Speech coding standards 32
3.5 Typical bandwidth requirements for some commonly used video formats 33

Glossary

AF Assured Forwarding. A type of per-hop behaviour for DiffServ aggregates of flows.

ARQ Automatic Repeat reQuest.

BER Bit Error Rate.

CBR Constant Bit Rate.

DCT Discrete Cosine Transform.

DiffServ Differentiated Services. A layer-3 approach to providing QoS to aggregates of flows.

DoS Denial of Service.

EF Expedited Forwarding. A type of per-hop behaviour for DiffServ flows.

FEC Forward Error Correction.

HDTV High Definition TV.

H.323 An ITU family of standards for IP-based videoconferencing.

IntServ Integrated Services. An approach to IP QoS that introduces services to provide finegrained assurances to individual flows.

ITU International Telecommunication Union.

MBone Multicast backBone. A multicast-enabled IP overlay network.

MS ASF Microsoft Advanced Streaming Format. An open video streaming format developed jointly by Microsoft, Real Networks, Intel, Adobe, and Vivo Software Inc.

MC Motion Compensation. A technique used in video encoding to reduce temporal redundancy in video and achieve higher compression rates.

MCU Multipoint Control Unit.

MNB Measuring Normalised Blocks. An objective speech quality assessment method developed at the Institute for Telecommunications Sciences.

MOS Mean Opinion Scores. A method of subjective quality evaluation of encoded multimedia content based on the collection and statistical manipulation of several quality ratings obtained by human subjects after viewing the corresponding material in a controlled environment.

MPEG Moving Picture Experts Group.

MPLS MultiProtocol Label Switching.

NFS Network File System.

PAMS Perceptual Analysis Measurement System. A speech quality metric developed at British Telecommunications.

PESQ Perceptual Evaluation of Speech Quality. A model of speech quality jointly developed by KPN Research and British Telecommunications.

PHB Per-Hop Behaviour.

PLR Packet Loss Rate.

POTS Plain Old Telephone Service.

PSNR Peak Signal-to-Noise Ratio.

PSQM Perceptual Speech Quality Measurement. A method of objective quality assessment of speech signals developed at KPN Research that is also an ITU-T Recommendation (P.861).

PSTN Public Switched Telephone Network.

RSVP resource ReSerVation Protocol. A signalling protocol used by the Integrated Services QoS model to establish quality-assured connections.

SDTV Standard Definition TV.

SNR Signal-to-Noise Ratio. A simple objective metric to measure the quality of a signal.

VoD Video on Demand.

VoIP Voice over IP.

VPN Virtual Private Network.

Chapter 1

Introduction

1.1 What are the advanced applications?

Today the Internet is predominantly used by conventional TCP-oriented services and applications such as the web, ftp, and email, enriched with static media types (images, animations, etc.). For the last few years, the Internet has also been used to transport modest-quality streaming audio-visual content. The Internet is also being used as a low-cost interactive voice and video communication medium. However, people are starting to realise the potential of using the plethora of already existing applications used in different disciplines and contexts over the Internet. We are seeing the emergence of a new generation of applications that can revolutionise the way people conduct research, work together and communicate. We call this new breed of applications *advanced Internet applications*. Advanced Internet applications can offer new opportunities for communication and collaboration, leverage teaching and learning, and significantly improve the way research groups are brought together to share scientific data and ideas. The use of advanced applications will facilitate new frontier applications that explore complex research problems, enable seamless collaboration and experimentation on a large scale, access and examine distributed data sets, and bring research teams closer together in a virtual research space. Advanced applications also involve a rich set of interactive media, more natural and intuitive user interfaces, new collaboration technologies using high quality sensory data, and interactive, real-time access to large distributed data repositories. To mention only a snapshot, application areas include:

- Interactive collaboration with high quality multisensory cues
- Real-time access to remote resources, like telescopes or microscopes
- Large-scale, multi-site scientific collaboration, computation and data mining
- Shared virtual reality
- Data Grid applications

Data and media flows of advanced Internet applications make great demands on all the components and devices on the end-to-end path. These are requirements for real-time operating system support, new distributed computing strategies and resources, databases, improved display and hardware capabilities, development of efficient middleware, and, of course, capabilities of the underlying network infrastructure. Large-scale scientific exploration and data mining require the exchange of large volumes of data (in the order of terabytes and petabytes) between remote sites. High quality data visualisation applications, videoconferencing and High

Definition TV (HDTV) demand huge amounts of bandwidth, often with tight timing requirements. On the other hand, there are applications that are highly sensitive to any loss of data. In order to function with acceptable quality, such applications require exceptionally high bandwidth, and also specific and/or bounded network treatment with respect to other network performance parameters (delay, jitter, loss, etc.). In other words, they require bounded worst-case performance, something that is generically called *"Quality of Service"*. As a best-effort Internet does not support any means of traffic differentiation, it cannot guarantee quality of service.

1.2 What is quality of service?

Quality of service is a very popular and overloaded term that is very often looked at from different perspectives by the networking and application-development communities. In networking, "Quality of Service" refers to the ability to provide different treatment to different classes of traffic. The primary goal is to increase the overall utility of the network by granting priority to higher-value or more performance-sensitive flows[1]. "Priority" means either lower drop probability or preferential queuing at congested interfaces. QoS that attempts to elevate the priority of certain flows above the level given to the default best-effort service class, requires admission control and policing of those flows to prevent theft of service. Such elevated services may provide hard worst-case performance assurances to certain flows. Non-elevated forms of QoS like Scavenger ***[cite]*** and ABE ***[cite]***, however, do not require policing, but provide applications a useful means to volunteer "hints" to the network about their needs. In either case, it should be noted that QoS does not prevent congestion; it merely adds "intelligence" at congested interfaces, allowing the network to make informed decisions about how to queue or drop packets.

In contrast, the view of QoS that application developers and application users often have, is more subjective. QoS is seen as something that will improve *my performance*. This is flawed and oversimplified. QoS may or may not improve an individual application's performance; results are highly dependent on the idiosyncratic relationship between a particular application's *utility* and the network performance it experiences[2]. The term "utility" is an umbrella term. It embraces perceived quality, that is, how pleasant or unpleasant the presentation quality is to the user of the application (e.g., visual quality of a displayed video sequence). Additionally, it may reflect the application's ability to perform its task (for example, in IP telephony, whether or not good conversation is achieved) or generate user interest (which in turn, may produce revenue — an important incentive). It is crucial to understand the relationship between application utility and network performance. In some cases, application performance objectives may be met either by increasing application sophistication (thereby reducing sensitivity to poor network performance) or by engineering the network to support QoS assurances (thereby guaranteeing that the application will not experience poor network performance). In other cases, applications and the network might share the burden, each becoming somewhat more sophisticated to improve overall utility in a cost-effective manner. Understanding these engineering tradeoffs is essential if application designers and network engineers are to make informed decisions about where to add money, effort, and complexity to meet the shared objective of enabling new Internet applications in a scalable and cost-effective manner.

Performance attributes are sometimes assigned different intepretations by different communities. For example, in networking, the term *delay* expresses the amount of time it takes for a data unit to propagate through the different paths of the network. For an application developer, e.g. a video system designer, delay is the time

[1] A flow can be defined in a number of ways. One common way refers to a combination of source and destination IP addresses, source and destination port numbers, and a session identifier. A more broad definition is that a flow is the set of packets generated from a certain application, interface or host. There is a debate on what is the appropriate granularity of a "flow", but nevertheless, each of the above definitions can be valid in the right context.

[2] It is also dependent on whether a particular individual can afford the extra cost of priority treatment. In many internet service markets, the cost of such priority treatment exceeds the cost of upgrading to faster best-effort service.

that is required for data to be encoded/decoded. It is very often the case that the two communities disregard the importance of this difference in perspective. For example, until recently the image processing community assumed that the underlying transmission infrastructure provides a reliable transport medium, a circuit-switched equivalent, in which the only delay is the propagation time and losses are rare and corrected by the physical or data-link layer. Thus they strived to maximise the quality of the encoded material by optimally selecting appropriate encoder/decoder parameters. In an transmission environment like the Internet, this assumption does not hold. For example, packet loss may dramatically degrade the quality of the encoded stream, and the perceptual distortion caused is usually far beyond that introduced by encoding artifacts. It is imperative that these misconceptions are corrected and that research communities achieve a shared understanding of what quality stands for and how it is affected.

1.3 The need to classify applications' requirements

There is a widely-held belief that advanced applications cannot be entirely accommodated by today's Internet, and that is necessary to have a service model that offers QoS guarantees to flows that need them. There is another camp that claims that QoS needs of applications can be sufficiently met by an over-provisioned best-effort network, combined with application intelligence to adapt to the changing availability of network resources and to tolerate loss and jitter. Both of the approaches have merits and disadvantages. It is probably true that an efficient solution lies somewhere between these two positions and favours some form of traffic differentiation.

It is apparent that without any form of traffic classification and prioritisation, network congestion will become a problem, affecting QoS-sensitive flows and reducing the quality of the corresponding applications. However, the selection of a suitable network model is a complicated function of several factors, such as the criticality of the applications, the complexity and scalability of the solution, and the economic model or the market needs. A very important factor is the kind of applications that are designed and expected to run over a network. Since networks are ultimately used by users running applications, it is imperative that the designers of networks and Internet service providers consider the effect of those applications operating over the network, and also the effect of the network's capabilities or service model on the usability and quality of applications. Network research, design, development, upgrades and configuration have to be carried out with the target applications' needs and requirements in mind. The reverse also holds. Applications need to consider the capabilities and limitations of the networks that are used to transmit their data. Applications that are unresponsive to network conditions can cause network congestion or even congestion collapse [37], reduce network utilisation, and suffer the consequences of their own behaviour.

Understanding the performance needs of advanced applications is essential, as it can provide both the network and applications R&D communities with a better understanding of how network services can be tailored to suit demands of advanced applications, and how advanced applications can exploit existing or new networks in a beneficial manner. Understanding application needs can allow applications to deploy built-in mechanisms that allow them to function with acceptable quality even on a network that at times displays characteristics that are far from ideal. In order to do so, it is necessary that the whole range of operational behaviours of applications be carefully explored and translated into proper adaptation mechanisms or policies. Such mechanisms and policies are particularly important for the application itself, as they will allow it to function in a wide range of networking environments, thus increasing its acceptance or marketability. The well-being of the underlying network will be also preserved.

This report is a working document; it should not be considered complete and exhaustive, but will be continually updated. Its purpose is twofold. First, the document investigates the QoS needs of Internet applications and the ranges of values of network performance metrics within which advanced applications operate with high or acceptable quality. This is how the application "expects" or "needs" to be treated by the network. Second,

the document investigates what behaviours an application can develop in order to (i) get the most out of the underlying network that transports its data flows, and (ii) in turn, "honour" the network and "protect" it from undesired circumstances. Both these issues are central to the success of advanced Internet applications and indicative of the need for closer cooperation between the application and the network, cooperation that needs to be further promoted. Good end-to-end application performance should become a task shared by the network and the application, seeking the best balance among network engineering, application design and economic incentives.

Chapter 2 presents a multidimensional taxonomy of Internet applications and investigates how this taxonomy relates to application performance characteristics and requirements. We present a high-level review of different classes of applications. Chapter 3 examines the issue of application quality and presents a detailed review of end-to-end performance requirements for two classes of applications: interactive IP audio (VoIP) and Internet video streaming and conferencing. In the last part of the chapter, we present an overview of adaptation techniques that audio and video applications may use to share the burden of QoS with the network. Chapter 4 discusses recent advances in the research and development of tools and methods for measuring application quality. This chapter focuses on quality assessment methods for audio and video. Finally, Chapter 5 concludes this report.

Chapter 2

Taxonomy of advanced applications

In this chapter we present a multi-dimensional taxonomy of advanced applications. Advanced applications display characteristics and features that do not occupy the same conceptual space, and it is therefore not feasible to define a taxonomy on a single dimension. Applications can be identified as belonging to one or more categories. This division can be based on the task they perform (task characteristics), the type of media they involve, the situation of operation (e.g., geographical dispersion of users) and the behavioural characteristics of users (e.g., user expectations, skills, etc.). The utility of an application, defined as its ability to successfully complete its task or as the quality perceived by the end user, is a function of all the above factors and is in many cases hard to define. In order to gain a better understanding of how the characteristics of an application define or dictate its quality requirements, we attempt to classify applications by examining their properties along the above-mentioned dimensions: task characteristics, type of media, user behaviour and situations of usage.

2.1 From application characteristics to application requirements

In this section, we present a first taxonomy or, more precisely, a grouping of advanced applications by considering common inner characteristics of applications and usage scenarios from a number of different viewpoints. For each class of applications, we try to devise generic, high-level guidelines for the specification of quality requirements, considering the fact that an application's behaviour is influenced by multiple factors.

2.1.1 Application task-centric classification

Applications can be categorised by considering the task they try to achieve or the kind of activities that take place. At a high level, application tasks can be classified into *Telepresence* and *Teledata*, and into *Foreground* and *Background* tasks, a division derived from Buxton [25].

Telepresence vs. Teledata. The distinction here is between applications that support communication, enable awareness between users and facilitate immersiveness in virtual environments (telepresence) — such as videoconferencing and virtual meetings — and applications that carry useful data to the user (teledata) — such as video or music streaming. In general, telepresence tasks can be identified as human-to-human tasks, while teledata involves human-to-machine interaction. In certain applications, both telepresence and teledata tasks may coexist. Activities that involve interaction between users will have different requirements than human-to-machine tasks, as the nature of interaction involves various user behaviours that affect quality differently.

More precisely, the definition of (tele)presence may differ depending on whether it is defined in the context of *virtual reality* or of *videoconferencing*:

> Presence in VR is usually defined as "being in" or "being part of" a mediated virtual environment, one that is different from the physical environment in which the observer is located. (Note: the existence of other people or their graphical representations is not relevant here.) The term "telepresence" introduces the notion of being in a remote location; as all environments in VR are virtual, the factor of remoteness is not applicable to them, and therefore the use of the prefix "tele-" is not appropriate. On the other hand, one could say that "telepresence" signifies being in environments that are shared by users who are at different geographical locations. They are now able to experience this shared virtual environment as the place where they all meet and interact. In VR literature, the correct term for this is "co-presence" or "social presence". In the case of videoconferencing systems, the term "telepresence" indicates the sense of being in a remote place. There is also another component of its definition: telepresence includes and assumes the existence of other people and interactions among them, something that is not part of the definition of presence in VR. Both definitions are correct in their respective areas; nevertheless, since a substantial part of this document deals with videoconferencing systems, to avoid misunderstanding we use the definition of telepresence as it is used with respect to these systems.

Foreground vs. Background tasks. The classification of an application task as foreground or background has major implications for how users perceive its quality. According to Buxton [25], a foreground task gets the full attention of the user, where a background task does not. Background tasks take place in the "periphery", usually introduced to promote or enable awareness. Foreground tasks are those that involve the user interacting with, monitoring and/or responding to ongoing activities; in background tasks, the role of the user is that of a passive observer. It is clear that foreground tasks will have significantly higher quality requirements than background tasks, and that background tasks can be accommodated with a modest set of resources that secures a low level of quality. Although background tasks or data are not tightly related to QoS requirements, they are still important in the context of advanced applications, as their absence would influence the way foreground tasks or data are perceived. For example, the lack of background noise in environments where it would naturally be expected to be present (street noise when you're wandering in a virtual city, or people noise in a virtual museum) will influence immersiveness, as it brings the user the feeling of a sterile and unnatural environment. This affects the application quality, in this case the feeling of presence in a real environment.

The above two divisions are orthogonal classifications and divide applications into four main types: foreground teledata, background teledata, foreground telepresence and background telepresence. Based on this classification of Buxton [25] we try to identify applications in each of these categories, and later, to outline generic quality requirements:

- **Foreground teledata.** Applications in this category include tasks that require the user to directly or indirectly access, monitor, manipulate or react to data, without requiring interpersonal communication between users (human to computer interaction). The data can be auditory, visual, haptic, olfactory, tracking, database and event transactions and synchronization, simulation, remote rendering, control, etc. Furthermore, the nature of one kind of data can be diverse (in the case of visual data it may change from streamed video to visualised datasets within an immersive environment). Thus the quality requirements for these applications will depend on the exact type of the application. For example, safety-critical applications, like remote surgery operations, will require far better quality performance for the video compartment than will viewing entertainment material (e.g., a music video clip). Furthermore,

the transmission and reception of non-sensory data, like instructions for remote devices, exhibit tight deadlines and require timely and lossless delivery if coordination the between actions and responses is to be maintained. In contrast, in cases where the auditory information is more important for the task of the application, like in a clip from a talk show, the users may be ready to accept lower video quality. In general, for an application dealing with video and audio sensory data, the acceptability or quality of the application will be affected by the relative importance of the visual and auditory components to understanding the message or meaning [9].

Usually this kind of application will require high audio-visual quality, because in such cases quality assessment depends not only on human factors but also on the criticality of the application. Also, as discussed above, the type of application will dictate whether certain data flows (e.g., data that control remote devices) have to be treated preferentially.

- **Background teledata.** Applications that deal with data that do not require direct manipulation by the user, but exist to create a certain awareness, effects, etc., like web cameras or ambient background sound, belong to this category. The quality requirements are obviously much lower, and such applications are out of scope for this document.

- **Foreground telepresence.** This includes all situations that require interaction and some kind of communication between human users (human-to-human). Human-to-human interaction can be achieved by means of sensory data like audio and video (videoconferencing), through avatar representations in a VR space, through a combination of both, and even through transmitting force feedback. Due to the interactivity requirements that these applications exhibit, they are particularly sensitive to end-to-end latency and delay variation. Network drop rate should be kept low, although there exist sophisticated techniques to alleviate the effects of packet loss (Forward Error Correction (FEC) or retransmission) at the expense of some extra latency.

  In telepresence applications, the auditory channel is quite important and in many cases the most vital means of communication. The existence of an auxiliary video channel improves the users' perception of the task. However, very low video frame rates (<5Hz) [79] can cause a mismatch between the auditory and visual cues leading to complete loss of lip synchronisation, with annoying perceptual results. For the visual feed to contribute to the application task and complement the audio, video frame rate needs to be over 15–16 Hz (or frames per second) [55, 15].

- **Background telepresence.** In this category of applications, a high level of interaction is not a requirement. These tasks will allow users to experience a "passive" awareness of other users' activities. Such tasks include low-frame-rate video feeds of silent participants (audience) in a videoconference. Like background teledata, these tasks do not have extreme requirements (e.g., a low frame refresh rate is adequate) and can be accommodated with far less resources; thus they will not be studied in this document.

We observe that background data do not possess particularly stringent network QoS requirements, as they are usually low-bandwidth flows that serve auxiliary purposes. If data transmission prioritisation is utilised, they can be low-priority flows or may be dropped, as they have less importance for the quality of the application (in comparison with foreground data). For these reasons, we subsequently ignore background tasks or data.

Interactive vs. non-interactive tasks. In interactive tasks, actions are followed by appropriate responses. Interactivity may arise between persons (interpersonal), between a human and a machine (e.g., remote instrument control), and between machines (machine-to-machine — e.g., data transactions). The degree to which the task is interactive is particularly important as it may determine the levels of tolerance to delay, jitter, etc.

Interactive applications will usually pose more stringent requirements than non-interactive ones, because of the promptness of response that is required.

Depending on the number of application users, interactive applications can be further divided into those involving *group-to-group* interactions and those involving *individual-to-individual* interactions. Naturally, group-to-group interactions are far more complicated, as they often involve large numbers of participants and teams working together (multiple sites and multiple participants per site).

The major difference between telepresence and teledata in terms of network QoS can be summarised as follows: the main aims of a telepresence application are to enable an environment for coherent remote communication and collaboration, and to create the feeling that the participant/subject is in a remote environment rather than his/her actual location. A telepresence application needs to preserve the main aspects of communication: good interactivity, and sensory cues that successfully serve the application task. This translates into a need for short latencies that can keep tasks synchronized (full-duplex conversation, joint operation of instruments, exploration of data, etc.). As interactivity poses low latency constraints, jitter needs to be kept controlled as well. For non-critical applications, the fidelity of the flows involved can be traded off, as long as the minimum requirements to achieve application tasks are met. Furthermore, packet loss can have a significant cost in quality degradation, especially since error protection and retransmission techniques are sometimes too time-expensive to be used. In the case of interactive teledata applications (remote surgery, remote control of telescopes/microscopes, etc.), interactivity requirements can be similar to, or even tighter than, those of interactive telepresence.

For non-interactive teledata applications, the constraints on latency and jitter can be more relaxed. One-way delays can be on the order of seconds without compromising application quality or causing user distraction/annoyance, and receiver de-jittering buffers can allow for comparatively high jitter values. On the other hand, there is an expectation of high quality media, mainly due to the nature of the application (e.g., music, entertainment video), thus the ability of these applications to adapt their transmission rates without sacrificing expected quality is more limited. Relaxed latency requirements also mean that sophisticated error correction and retransmission can be used to reclaim data corrupted due to loss.

Machine-to-machine tasks. The above mentioned tasks are related to interactions between humans, or between humans and machines. There are also machine-to-machine applications that do not involve any human intervention or interactivity as part of their operation. Typical scenarios include the transmission of data among various computers, manipulation and processing of data, creation and transmission of transaction data, distributed computing, and exchange of control data. Not all of these computer-to-computer tasks will require high levels of network service. With this type of application, quality requirements are dictated not by human factors, but rather by the exact properties of the application, such as delay sensitivity, requirements for timely delivery (e.g., time- or safety-critical applications), or volume of data to be exchanged between remotely located nodes. End-to-end delay and delay variation are the most crucial performance parameters for those applications that transfer control data. Throughput is important for applications requiring large data transfers.

Figure 2.1 graphically outlines task-based categorisation of applications.

2.1.2 User characteristics

Specific user characteristics influence the quality requirements of applications to a great degree. Some of these factors are:

- Users' familiarity and experience with the application. Experienced users tend to overcome some difficulties more easily, as they can adapt to the medium and use the application more effectively than novice

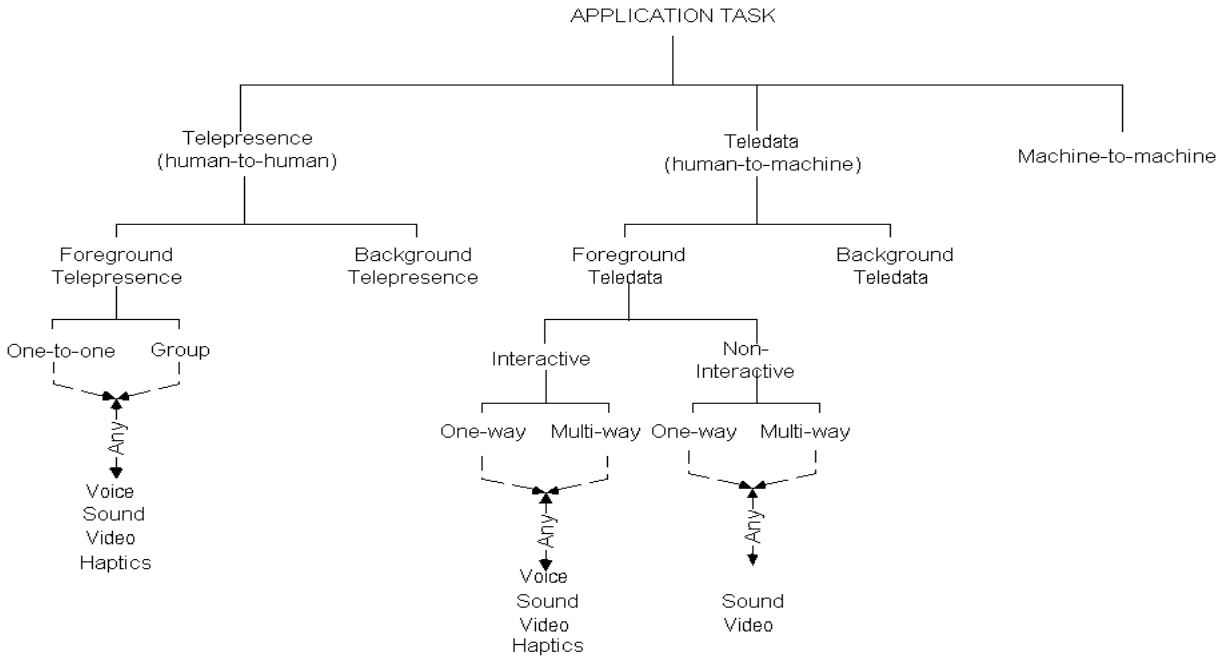

Figure 2.1: A taxonomy based on the task of the application ***(incomplete...)***

users. They may also have a better understanding of the application modules and may use them more efficiently to manage a task. Familiarity — contact of humans with actions, situations or persons over a period of time — is another factor that can influence users' performance of tasks. For example, in a teleconference, if the peer speaker is someone familiar, then communication is easier, even if the quality of the media degrades below what would usually be considered an acceptable level.

- Users' expectations. This is a very important aspect of user behaviour that we believe is central to determining the quality requirements of an application, and that can be used to explain the (sometimes unpredicted) fact that users exhibit tolerance to a high degree of impairments in some cases but not others. A user is satisfied if the service she receives from the network and application meets her expectations (predictability). Such expectations are determined by experience with the use of similar services in a different environment (e.g., mobile telephony), economics (e.g., a service being considerably cheaper than available alternatives), or lack of alternatives ("this is the only way I can watch the game").

- For conferencing telepresence applications, the fluency of the participants in the language(s) used during communication is also very important. Whether or not all speakers have fluency in the spoken language makes a big difference to the application requirements. If so, communication is more straightforward, even if the audio and video signals degrade. If not, clear voice, that is, high quality of the transmitted voice signal, is necessary. Furthermore, better communication can be achieved if voice is accompanied by good-quality, lip-synched video.

- Number of users (or participating nodes) and their geographical distribution or remoteness. The number of participants in an application scenario contributes to the application's behaviour over multiple dimensions. Data-intensive computation or experimentation generates and exchanges large amounts of data, thus increasing the application's aggregate bandwidth requirement. In collaborative applications,

a large number of remote participants increase the need for conference control as well as the aggregate bandwidth of the session. Wide geographical distribution means that there are more extreme and diverse latencies (round-trip times) for the different participating sites, which makes conversation or synchronisation of actions more difficult to manage, and also makes the tasks of network transport layer services (e.g., congestion control) more difficult.

- Other factors, like the user's age, background, sensory dominance, or hearing or sight disability.

We can therefore comprehend that the behavioural patterns and expectations of the users can provide a solid base for the extraction of generic requirements for an application. While user-related behaviour is highly subjective, we can usually recognise the profile of a typical user and design the application service in accordance with the preferences, requirements and expectations of that typical user.

2.1.3 Elastic, tolerant, and adaptive applications

In this paragraph we examine application properties attributed to the nature and transport requirements of the participating media flows.

Elastic vs. inelastic. *Elastic* applications can tolerate significant variations in throughput and delay without considerably affecting their quality; as network performance degrades, application utility degrades gracefully. These are traditional data transfer applications like file transfer, email and some http traffic. While long delays and throughput fluctuations may degrade performance, the actual outcome of the data transfer is not affected by unfavourable network conditions. However, certain constraints may arise when these services are considered in the context of advanced applications. Elastic traffic can be further categorized by delay and throughput requirements:

- *Asynchronous bulk traffic*, such as email and voice mail. Latency and throughput requirements are very relaxed; hence, these do not constitute advanced applications in the context of this document.

- *Interactive burst traffic*, like telnet, or Network File System (NFS) traffic. These interactive applications require near-real-time human-to-machine interaction and ideally should have delays of 200–300ms or less, but could possibly tolerate slightly more.

- *Interactive bulk traffic*, like file (ftp) and web (http) transfers. In general, interactive bulk applications will also have similar requirements for delay, but they will generally prefer high throughput. In most cases, users are willing to tolerate delay that is roughly proportional to the volume of data being transferred. Web access is an interactive, non-real-time service. However, the interaction is important and fast responses are required. In the context of advanced applications, like large-scale scientific exploration, data mining and visualisation or Grid applications, the volume of data and the requirement for real-time (or near-real-time) data pre- or post-processing means that much higher throughput and tighter latency constraints arise.

Inelastic applications (also called *real-time* applications) are comparatively intolerant to delay, delay variance, throughput variance and errors, because they usually support some kind of QoS-sensitive media like voice or remote control commands. If certain QoS provisions are not in place, the quality may become unacceptable and the application may lose utility. However, depending on the application task and the media types it involves, the application can successfully operate within a range of QoS values. For example, audio and video streaming applications, being very loosely interactive applications, are not extremely sensitive to delay and jitter.

Tolerant vs. intolerant. Some inelastic applications can tolerate certain levels of QoS degradation (*tolerant* applications) and can operate within a range of QoS provisions with acceptable or satisfactory quality. A video application can tolerate a certain amount of packet loss without the resulting impairments becoming significantly annoying to the user. Consequently, tolerant applications can be:

- *Adaptive.* Tolerant adaptive applications may be able to withstand certain levels of delay variation by building a de-jittering buffer, or adapt to available bit rate, packet loss and congestion by gracefully reducing their encoding or transmission bit-rate (e.g., a video stream can drop a few packets, frames or layers). Adaptive applications are to some degree capable of adjusting their resource demands within a range of acceptable values. Application adaptation is triggered by appropriate mechanisms that directly or indirectly inform the application of the current network performance. Adaptation is also very important in the context of Internet congestion control. An application can adapt to packet loss, which is primarily an indication of network congestion, by responsively reducing the transmission rate.
- *Non-adaptive.* Tolerant non-adaptive applications cannot adapt in the same fashion, but can still tolerate some network QoS variation. For example, the quality of an audio or video flow may be degraded by loss, but still be intelligible to the user, listener or viewer.

On the other hand, there are applications that fail to accomplish their tasks sufficiently if their QoS demands are not met. These applications are called *intolerant.* An example of such an application is remote control of mission-critical equipment, such as a robot arm or surgical instruments. Some applications may be able to adapt their rate to instantaneous changes in throughout (rate-adaptive), while others may be totally non-adaptive.

2.2 A taxonomy based on type and interdependencies between media

Before identifying specific emerging applications, we first take a look at the building blocks of applications. These are lower level, sometimes even *abstract* applications that are uniquely identified to reside in a number of applications. For example, a two-way audio module is present in a number of specific applications, such as videoconferencing and collaboration within virtual environments. These "generic" applications will preserve certain QoS requirements within the context of the higher-level application, thus making them a starting point in the study of the application's behaviour. However, the context of the application within which the specific generic applications operate will probably alter their quality characteristics or importance, based on the task of the application. Moreover, interactions between application components or media within an application also pose extra requirements. In the following sections, we present, where applicable, the characteristics of such "basic" applications. We then discuss applications that involve one or more generic applications, study their behaviour and the interactions that arise, and look into the problem of how these interactions affect the QoS requirements of the application. It is important to note that when such interactions occur (for example, inter-stream synchronisation), the effect on perceived quality may be different than when we observe generic applications in isolation. Nevertheless, studying the behaviour of these elementary applications will enable us to get a good understanding of their needs.

2.3 Types of generic applications

We start our study by observing the behaviour and needs of 'elementary' applications. Some of these applications are not applications per se, but rather constitute building blocks of applications. However, from the network

point of view, they form autonomous modules, have their own transport mechanisms, and as such, pose specific demands to the underlying transport medium. Obviously it is not feasible to draw specific conclusions about the behaviour and quality characteristics of these elementary applications, because the context of the application within which they are integrated creates new interdependencies that cannot be described in such a simple way (synchronisation with other media flows, user-specific importance, etc.). Nevertheless, it is worthwhile to look at the characteristics of these applications as a baseline approach to our analysis. We attempt to analyse these demands in the following paragraphs.

- two-way or multi-way high-quality interactive audio, Voice over IP (VoIP)
- two-way or multi-way high-quality interactive real-time video
- high-quality audio and video streaming
- transfer of high resolution images and 3D graphics
- haptics
- bulk data transfer for data mining
- remote control of devices
- database and event transactions and synchronization

2.4 Classes of higher level applications

2.4.1 Auditory applications

2.4.1.1 Interactive

Conversational audio. Voice communication is still the dominant type of remote human communication. It can be characterised as a foreground, interactive, telepresence application. IP telephony has recently become a competitive alternative to Public Switched Telephone Network (PSTN), mainly due to its simplicity, economic viability and added value gained from a computer-aided telephony service. VoIP is probably the first advanced application that has significant deployment in today's Internet. In the industry, VoIP is becoming common for voice communication through corporate intranets, and as backbone capacities grow and service differentiation becomes available, telecommunication carriers most likely will rely on the Internet to provide telephone service to geographic locations that today are high-tariff areas. Since interactivity is the main requirement, low end-to-end delay and jitter are very important parameters in maintaining conversational quality, and low packet loss is needed to sustain the audio signal's quality. Resilience to packet loss is also dependent on the specific audio codec used (see section 3.2). The business case for VoIP can be made much more compelling by making VoIP a better experience than Plain Old Telephone Service (POTS) (e.g. better fidelity, better integration with messaging and presence).

For multi-way voice communication over the Internet, the Multicast backBone (MBone)[1] and the MBone tools[2] have been successful enablers of one-to-many and many-to-many communication for large groups.

Parenthetically, multicast transmission can be used to transport data between multiple sites efficiently and has been proved to be the only scalable solution for multi-scale, multi-user or large-audience applications. Such

[1] http://www.cs.columbia.edu/ hgs/internet/mbone-faq.html
[2] http://www-mice.cs.ucl.ac.uk/multimedia/software/

applications include the broadcast of popular live or recorded events, broadcast-style Internet TV, group communication, and distribution of data. Despite its efficiency, IP multicast still retains the problem of insufficient adoption by network operators. The important issues that need to be resolved in order for multicast services to be widely deployed are how multicast will interact with traditional IP unicast (specifically, issues of multicast congestion control), fairness between unicast and multicast traffic/flows, and support of service differentiation in IP multicast.

High quality audio orchestration. Audio orchestration imposes stringent timing requirements on participating audio flows. Such applications involve geographically dispersed sources of high-quality, multi-channel sound that need to be orchestrated with tight timing (e.g., a tele-concert) in order to maintain synchronisation, and have completely different requirements from the audio streaming case we examined above. Due to the stringent requirements, end-to-end delay and jitter are crucial factors. High quality expectations mean that multi-channel audio streams may have to be transmitted uncompressed, which increases the demand from the network service in terms of sustainable bit rate. Furthermore, human factors indicate that users are far less tolerant of quality degradation of entertainment sound or music. Thus, besides sustaining the required network bit rate for the audio stream (e.g., $\geq 128Kbps$ for stereo MP3, or $> 1.5Mbps$ for six-channel AC-3 Dolby sound), packet loss should remain very low if we want to preserve audible quality.

2.4.1.2 Non-interactive or loosely interactive

Professional-quality audio streaming. High quality means high-sampling (16- or 24-bit samples), multi-channel audio (up to 10 channels or more) with CD-equivalent or better quality (e.g., 96KHz sampling rate). In order to maintain the high quality of the original signal (for example, in remote music recording or music distribution scenarios), the streams might need to be transmitted uncompressed or losslessly compressed. However, audio streaming may have completely different application requirements. Since interactivity is not a constraint[3] (for example, a user is willing to wait for a while, even on the order of seconds, before the music starts playing), the application might be able to build up de-jittering buffers and thus tolerate a certain amount of delay and jitter. Furthermore, less sensitivity to delay means that certain error correction algorithms can be employed to increase the robustness of the stream to packet loss. Robustness to loss can also be enhanced by means of retransmission. If such techniques are not employed, then packet loss should be kept at very low levels. This leaves one major requirement for high-quality streaming applications: a sustainable bit-rate. However, there are adaptivity opportunities within these applications. A number of adaptation choices can be made in order to restrict the transmission rate of the audio stream to adjust to the available (nominal) network bit rate — layering, dropping of transmitted channels, transcoding, etc. — but this entails some degradation in quality (see section 3.5.1).

2.4.2 Video-based applications

2.4.2.1 Interactive

High-quality audiovisual conferencing. Until recently, videoconferencing required expensive equipment, specialised room setup and complicated conference control. PC-based conferencing equipment is now an affordable commodity, and H.323-based IP conferencing can be easily set up and maintained. Currently, Internet videoconferencing is restricted to low/modest bit rates (300–500Kbps), precluding a high-quality communica-

[3]Some interaction may still remain, e.g., for play, pause, stop, or other player-control actions

tion experience[4]. If the appropriate network resources are in place, high-quality videoconferencing can provide an exciting means of collaboration.

The term "videoconferencing" is loosely used in this section to include all foreground, interactive, tele-presence applications that involve high-quality audio and video to enable communication, education, distance learning, entertainment, tele-medicine or collaboration between remotely located individuals. Depending on the exact nature of the conferencing application and what type of media and media encodings are present, the quality requirements may vary accordingly:

- High-quality collaborative videoconferencing. These applications can be used for collaboration between remotely located users, presentations, project meetings, and meetings within enterprise intranets. They involve the transmission of high-quality audio and video streams (and possibly others, like high quality graphics and virtual environments) to offer an advanced collaboration experience.
- Video and audio orchestration. Similar to audio orchestration, with the inclusion of real-time video. Such applications can present artistic events, like dance performances and orchestras, or record music from distributed sources. As in the case of high-quality audio orchestration, such applications will require tight timing constraints on end-to-end latency and jitter to achieve a coordinated and synchronised performance.
- Medical surgery applications, tele-medicine and tele-diagnosis. These applications may have educational purposes (e.g., high quality video of the surgery and audio narration from the surgeon can be transmitted to a remote audience of medical students), or may enable a skilled surgeon to remotely operate with the help of live video feedback and haptic controls. In order to achieve good synchronisation with the surgeon's haptic equipment, the requirements for such critical applications are for good fidelity (at least VHS-quality, e.g., >1Mbps MPEG-1 video) video and low latency and jitter.
- ***Remote learning applications.***

Videoconferencing technologies There are several different modes of videoconferencing spanning all ranges of quality: from quarter-screen images with low-to-modest quality (less than VCR quality) to broadcast-TV quality and higher [53]. Videoconferencing can be used with desktop or room-based environments and may involve tight (scheduled and controlled join) or loose (join-at-will) conference control. It should be noted that this type of application is not immersive. The user experiences either a head-and-shoulders view of the current speaker (determined by some means of floor control) or a "Hollywood Squares" grid of all participants. Some of the most popular videoconferencing technologies include:

- *H.323* is an International Telecommunication Union (ITU) standard for videoconferencing, and is used for both point-to-point and multi-point teleconferences. For point-to-point conferences, users dial up other users, just as they would if they were making a phone call. For multi-point conferences, all participants connect to a common address at an Multipoint Control Unit (MCU), which acts as a reflector for all audio and video streams. The H.323 standard specifies the use of a standard video encoding format, H.261, and an optional H.263 video codec. Additional audio and video encoding formats can be used, including higher-quality, higher-bandwidth formats such as MPEG-1 and MPEG-2. Popular H.323 hardware vendors include Polycom, VCON, Picture-Tel, and Radvision.
- The *MBone tools* [80] are a collection of audio, video and whiteboard applications that use the Internet multicast protocols to enable multi-way (point-to-multipoint and multipoint-to-multipoint) communication. The MBone tools provide a variety of audio and video codecs.

[4]Currently, due to limited bandwidth and unpredictable behaviour of the Internet, typical videoconferencing suffers from relatively low video frame rates, lack of lip-synchronisation (time-lag between the audio and video components), modest quality of audio (e.g., mono) and low video resolution (usually QCIF or CIF) and fidelity

- The *Access Grid* [120] supports enhanced high-quality scientific collaboration among dispersed users, allowing real-time visualisation of data, distributed computation, human-computer interaction, meetings and training sessions. The main idea is to promote group-to-group rather than one-to-one collaboration by using underlying grid technologies and infrastructure. The Access Grid requires access to a number of resources — large multimedia displays, presentation facilities, interfaces to visualisation toolkits, interfaces to grid middleware — in order to facilitate next-generation scientific collaboration among distributed nodes. An AG node consists of multiple cameras, microphones and projectors within the meeting room space, as well as the software that enables natural interaction among the users. An Access Grid-enabled room contains video cameras covering several different angles and providing close and wide shots of the space. Wall-sized projection screens can show dozens of full motion video images from remote sites along with presentation materials (e.g., PowerPoint slides). Audio is enabled through microphones and speakers around the room. Based on the MBone tools, the Access Grid relies on network multicast and can put serious stress on campus routers. Even a modest-scale Access Grid event can generate over 20 Mbps of multicast traffic. Access Grid nodes use very high bandwidth (100s of megabits per second) Internet research networks and multimedia-intensive applications to enable real-time collaboration among users in different locations. AG nodes (the main components that support the Access Grid and provide all the above services) are being designed to enable support for mobile users[5].

- The Moving Picture Experts Group (MPEG) family of standards for encoding audio and video is another high-quality videoconferencing option. MPEG-1 provides near-VCR quality with typical rates between 1.5 and 3 Mbps[6] and MPEG-2 achieves broadcast-TV and higher qualities at rates of 5–30 Mbps and higher. However, cost of hardware (codecs[7], cameras[8] and microphones) is high. MPEG-4 is a new standard providing a wide range of quality and data transfer rates. Its initial commercial focus is video-on-demand streaming with quality higher than that of MPEG-1 but at only 300–400 Kbps.

When using videoconferencing technologies, there are several other issues to be considered:

1. ***Production values and cost.***

2. Delay. If higher compression is needed to accommodate more bandwidth-limited networks, then encoding delay may interfere with interactivity.

3. Interoperability issues, when there is a need to communicate between sites that use different technologies. Hopefully, gateways that enable this will soon exist for more of the technologies mentioned above.

4. ***Floor control, to allow coordinated sessions.***

5. Security. This is very important to promote the widespread use of videoconferencing in cases where privacy is a requirement.

6. Wide support of IP multicast. Not all networks support multicast, and MBone-based environments cannot become popular unless multicast is widely deployed.

For an outline of videoconferencing issues, refer to [53].

[5]The chosen platform is the Motorola iPAQ handheld, running Linux and using rat and vic multimedia software [80].
[6]MPEG-1 is mostly used for video streaming.
[7]There are no MPEG-2 decoders for videoconferencing, but only for streaming.
[8]Camera quality is very important for MPEG-2.

2.4.2.2 Non-interactive

Transmission of stored and live video is already very popular on today's Internet. However, it is confined to modest quality, low resolutions and restricted frame rates, due to the huge bandwidth requirements that higher-quality video acquires. These services are foreground, teledata applications, and as such will require significantly higher bandwidth, but have more relaxed latency requirements. The exact nature of the application and the geographical distribution of the users impose extra requirements, as discussed in section 2.1.2 and in the text below.

Video broadcast, streaming, video on demand. Video streaming falls into two categories: real-time dissemination of live events (broadcasting, or webcasting), and streaming of on-demand material. The former application scenario typically involves the transmission of events such as news, sports, remote experimental observations (e.g., eclipses), or rocket launches to significant numbers of viewers. Due to the viewer group sizes involved, such one-to-many applications are in practice best served in a scalable fashion by multicast networks[9]. The latter category is typified by client applications gaining access (one-to-one) to pre-recorded, stored video material from a remote server, for entertainment, training, education, etc. The common attribute of these services is that they do not, with the exception of *start/play/pause* actions, involve any high-level interaction or interpersonal communication. This means more relaxed demands for latency; jitter of tens of milliseconds can be alleviated with appropriate buffering algorithms and an initial delay to build de-jittering buffers. As mentioned earlier in the task-based application taxonomy (section 2.1.1), users of teledata applications require high quality, thus a certain level of sustained bandwidth has to be available to the applications. Furthermore, even though low loss rates are important to keep the quality distortions low, application tolerance of latency means that loss-protection techniques, like FEC, or re-transmission (Automatic Repeat reQuest (ARQ)), can be used to facilitate sustainable high quality even in the presence of higher network packet loss.

High Definition TV. HDTV is an application that might be able to stretch high-speed networks to the limits. HDTV [122] can provide high-resolution (16x9 aspect ratio, 1920x1080 (1080i), at 60Hz interlaced) moving images at qualities comparable to or better than any contemporary digital equivalent (like DVD). This, combined with high-quality surround sound, by far surpasses today's TV experience. Depending on the compression algorithm used, HDTV signals can be sent at rates well over 200 Mbps.

HDTV is available at different levels of quality for different target audiences:

- Consumer-grade (broadcast) quality, at 19.2 Mbps.
- Contributor quality, at 40 Mbps MPEG-2.
- Studio quality, HDCAM compressed at 200 Mbps.
- Standard Definition TV (SDTV), or SDI, that provides picture quality similar to that of Digital Versatile Disks (DVD) at 270 Mbps. This target is selected because lots of broadcast equipment is based on SDI.
- Raw (uncompressed) HDTV video at 1.5 Gbps using the SMPTE-292M standard [107]. This allows HDTV content to be delivered uncompressed through the various cycles of production. Here video content needs to be transmitted in uncompressed form because the degradation to quality caused by any introduced compression is undesirable for high-quality production use (studio production). Another reason for uncompressed HDTV is to reduce latency for high-quality interactive telepresence applications.

[9] However, there are still a lot of problems with native network multicast (e.g. pricing, security, scaling to large numbers of groups)

As can be seen from the above, HDTV is a major potential consumer of network bandwidth. From some preliminary experiments[10] that involved the transmission of 40 Mbps MPEG-2 and 20 0Mbps HDCAM/SDTV packet HDTV over an OC-12 Internet2 backbone network (Abilene), it was reported that the 40 Mbps stream with RAID-style FEC had minimal latency and could withstand 5-10% loss (depending on the decoder). For the 200 Mbps stream, buffering and retransmission were used for loss resilience; this resulted in a 4 second start-up delay and significant resilience to loss (10-15%). Given the huge demands of HDTV for bandwidth, multicast transmission seems the only scalable solution if HDTV is to be used for large-scale broadcast of audiovisual data.

2.4.3 Distributed Virtual Environments (DVEs)

Remote collaboration using traditional forms of media, like audio and video, offers a useful means of communicating. However, advances in 3D graphics rendering techniques, and increasingly powerful underlying hardware, give rise to far more exciting and diverse ways of supporting collaboration, scientific exploration, instrument operation and data visualisation, by supporting continuous media flows, data transfers and data manipulation tools within virtual environment interfaces. Other specialised equipment, such as haptic devices or head-mounted displays, is used to enable manipulation of instruments or navigation within virtual worlds. These collaborative distributed immersive environments impose new requirements on the network in terms of data transfer bit rates, short one-way delay for real-time operation, and reliable transfer of data.

2.4.4 Tele-immersion

The term "tele-immersion" describes a system and environment that provides the user with the illusion of sharing the same mediated space with other people who may be geographically distributed all over the world. This definition is valid for all research groups and their work in this area. However, the way different systems represent humans, the projection systems, and the user interfaces vary from one group to another. We will describe two typical approaches, one fostered by the National Tele-Immersion Initiative NTII[11], a research consortium made up of several US universities, and the other one associated with the CAVE community[12].

The system developed by the National Tele-Immersion Initiative uses 3D real-time acquisition to capture accurate dynamic models of humans. The system combines these models with a previously-acquired static 3D background (a 3D model of an office, for example) and it also adds synthetic 3D graphics objects that may not exist at all in the real world. These objects may be used as a basis for a collaborative design process that users immersed in the system can work on. The system can also be described as a mix of virtual reality and 3D videoconferencing. A very important part of the system, which consumes a great deal of processing power, is deriving 3D information from a set of 2D images, a task that the computer vision part of the system deals with. The advantage of this approach is that humans are represented very accurately — there is no need to model and simulate them, something that we still do not know how to do accurately. There is also no need to have models of humans prepared in advance — their 3D representations are obtained in real time as they enter the space that is covered by 2D cameras.

The downside of this approach is that 3D acquisition processing still takes a lot of computational power. As a result we have neither a real-time frame rate nor the resolution that one would like to have in systems that support human-to-human interactions. As processors and cameras become much faster, this will be less of a problem. Applications where this system would be superior are those that require very accurate representation

[10] Reported at the 2000 Internet2 QoS meeting in Houston — see http://www.internet2.edu/qos/houston2000/proceedings/Gray/20000209-QoS2000-Gray.pdf

[11] http://www.advanced.org/teleimmersion.htm

[12] http://www.evl.uic.edu/research/vrdev.html

of particular humans, their appearance and the way they move and gesture. The typical examples are tele-diagnosis and tele-medicine. As for the projection system, the NTII solution uses surfaces that are part of the regular working environment — walls in front of and around the office desk — which removes the need for specially designed rooms and expensive projection constructions. The entire system requires high bandwidth, minimal delay, and low jitter in order to send constant streams of 3D data that represent remote participants. The system is a classic example of a highly demanding and QoS-hungry application in the Internet2 environment.

The tele-immersive system that uses CAVE environments, on the other hand, uses avatars (3D graphical models and approximations of human bodies) to represent participants in a tele-immersive session. In order to achieve good results in mimicking humans, the system has to invest a lot of processing power to simulate the appearance, movements, and gestures of the human body. This is an extremely tough task if one wants to be as accurate as possible. So far there are no algorithms that do this to a degree that is completely satisfying for all the intricacies of human-to-human interactions. However, such accuracy may not be necessary in some applications, so certain approximations may be good enough. In the CAVE system all models of humans have to be prepared and available before the session starts. These models may be very different from the actual person who will be using the system. This effect may be desirable in some applications — creating an exact replica of an individual may not be the goal of that particular application. For the projection system the CAVE installation uses specially-designed multiple canvases onto which the imagery is projected. The entire system is still too expensive to be considered for massive deployment. It is important to note that this system is far less demanding in terms of the bandwidth required between the remote users (requirements for minimal delay and low jitter still exist).

Tele-immersive data exploration. Tele-immersive data exploration combines data queries from distributed databases, use of real-time or near-real-time tools to facilitate data exploration, and visualisation of data using immersive environments. It is basically a combination of data mining techniques with collaborative virtual reality. Tele-immersive data exploration applications will allow users to explore large and complicated data sets and to interact with visualised versions of the data in an immersive environment. Furthermore, they will allow collaboration among remotely-located users during the data exploration and visualisation process, as well as real-time data manipulation and data processing.

2.4.5 Remote control of instruments

The ability to reliably and promptly access remotely located instruments and devices is an enabler of great applications in the fields of science and medicine. For example, as the real-time control of instruments becomes practical, these techniques will be adopted by the medical community. Applications are envisioned in pathology, dialysis and even robotic surgery. For remote control tele-medicine, besides the real-time control data to be transmitted, high-quality video feedback and high-resolution images are also required. Other applications of remote instrument control include remote control of scientific instruments, such as telescopes and powerful microscopes.

This kind of application makes several demands on the network. As these applications involve a combination of real-time control and sensory data, they have stringent interactivity requirements.

- Reliability. This is the number-one absolute necessity for critical applications, e.g. remote surgery. For these applications, loss of any control/command data could be catastrophic and is unacceptable.

- As the main requirement is reliable and timely delivery of the control data that directs the remote instruments, having controlled end-to-end delay and minimal delay variation is absolutely crucial. Jitter control

is especially important, as jitter can lead to disorientation and erroneous responses on the part of the user of the remote control system, possibly leading to catastrophic or life-threatening situations.

- The bit-rate requirement depends on the participating data flows. Commands and remote control data are relatively low-bandwidth. In most cases, high-resolution images (images from telescopes or electron microscopes, X-rays or tomography images, etc.) and/or high fidelity video feeds will be transmitted from the remote location to the user, so a certain throughput will have to be sustained throughout the session.

2.4.6 Grid computing

Efficient scientific exploration and collaboration require the exchange of large amounts of data among scientific laboratories. Future research will be increasingly interdisciplinary, and thus specific data sets that are produced by independent labs around the world will need to be exchanged among a number of large-scale distributed sites supporting heterogeneous networks and computing and storage resources located around the globe. The objective of Data Grid projects is to enable next-generation, large-scale scientific exploration that requires intensive computations on large volumes of data (terabytes to petabytes) that are resident on numerous distributed large-scale databases. Applications of large-scale Grid computing cover a broad range of technology and science areas:

- particle physics and high-energy physics experiments
- atmospheric and earth observations for weather forecasting or monitoring of climate change
- experiments on complex phenomena of nature
- biological sciences
- astronomy and space observations
- collaborative engineering

Grid computing involves a plethora of applications, devices, distributed computing contexts, experimental tools, data manipulation software, large-scale database access and data exploration tools, services (archiving, security, caching, etc.), and VR tools, as well as transmission of continuous media flows and use of heterogeneous network and transmission platforms. Grid traffic includes bulk data transfer, high-priority access to remote databases, grid control traffic, exchange of audio and video for communication (videoconferencing) and visualisation purposes, and interactive data visualisation. Grid systems are foreground teledata and telepresence applications with different levels of interactivity. All these diverse traffic types have their own distinct QoS demands[13], and unless some data segregation mechanisms are used, they cannot achieve proper operation.

Grid computing projects represent the beginning of a new world of IT computation. One of the most attractive challenges, as identified by the Grid Physics Network project *GridPhyN* [46], is the realisation of the *Virtual Data* concept, which advocates the creation of a virtual data space consisting of large distributed datasets. Together with the development of global resource management techniques, policies, and security mechanisms, these constitute computational environments referred as Petascale Virtual Data Grids (PVDGs).

[13]For example, bulk data transfer will require big ftp pipes that will compete with delay- and jitter-sensitive traffic, like control traffic or continuous media flows.

2.5 Example applications and projects

In this section, we present a selected set of existing applications and projects that experiment with the use of advanced applications over high-speed, next-generation networks. Most of the information presented here can also be found in the Internet2 applications archive [14] which features many exciting networked application showcases and links to sources of detailed information for each of them.

2.5.1 Video-based applications

We review some of the most prominent video-based Internet2 projects. These projects promote research and collaboration through the use of high-quality media streams and flexible application architectures.

2.5.1.1 H.323-based videoconferencing

Megaconference. The *Megaconference* [121] project involves a series of high-quality videoconferencing meetings based on H.323. It is a "permanent, continuous multipoint H.323 video conference" hosted at Ohio State University and featuring several MCUs scattered around the world. Megaconference events consist of multiple sites collectively engaging in a live demonstration of the capabilities of H.323. Each participating institution/organization has an opportunity to address the conference participants, speak about their deployment of H.323, and showcase H.323 applications at their site. The event in 2001 was the largest H.323 multipoint conference conducted to date, and was simultaneously broadcast on the Internet in MPEG-1, Real and QT4 formats. The Megaconferences continue to approximately double in size every year. Megaconference I had about 50 sites, Megaconference II about 100 sites, and Megaconference III (2001) around 200 sites and 25 MCUs[15]. They are the largest Internet videoconferencing events and continuously push the state of the art in videoconferencing technologies and networking.

ViDeNet. ViDeNet [137] was created by ViDe to be a testbed and model network in which to develop and promote highly scalable and robust networked video technologies, and to create a seamless global environment for teleconferencing and collaboration. From a technical perspective, ViDeNet is a mesh of interconnected H.323 zones. Each zone represents a collection of users at each site that are administered by the site itself. ViDeNet enables end users registered with each zone to transparently call each other, thus facilitating seamless use.

ViDe LSVNP. The ViDe Large Scale Video Network Prototype (LSVNP) [92] is a distributed H.323 videoconferencing testbed, funded by the Southeastern Universities Research Association and BBN, the research arm of GTE. Its goals are to explore issues critical to the deployment of seamless networked video and to accelerate the deployment of H.323 through resolution of large-scale deployment issues. BBN is collaborating with ViDe to utilize the LSVNP to conduct analysis of video traffic patterns. The LSVNP testbed is the first large-scale distributed videoconferencing network. A number of projects are currently being supported with gatekeeping and multipoint services. The projects include applications in marine sciences, veterinary medicine, speech pathology and audiology, training for teachers, architecture, higher education outreach, technical assistance for people with disabilities (deafness), emergency telemedicine, and earthquake research.

VRVS. One MBone implementation example is the Virtual Rooms Videoconferencing System (VRVS) [133] from the California Institute of Technology and CERN, the European particle physics laboratory. With the

[14] http://apps.internet2.edu/html/archives.html
[15] http://www.mega-net.net/megaconference/finalreport.htm

objective of supporting collaborations within the global high-energy physics community, VRVS has deployed reflectors that also allow participation by non-multicast-enabled sites. In addition, the VRVS team has developed gateways that allow participation using non-MBone tools such as H.323, QuickTime, and MPEG-2. Most of the software used by the MBone-based environments is freely available (e.g., [80]), and can be used with low-cost conferencing equipment (desktop cameras, microphones, etc.).

More information on the above can be found in the 'Videoconferencing Cookbook' from the Video Development Initiative [131].

2.5.1.2 Music video recording

Music video recording via Internet2. This was an Internet2 project[16]. The goal was multi-location music video recording using real-time streaming video over Internet2 networks. The participants included NYU, USC, U. Alabama-Birmingham, U. Miami and U. Georgia School of Music. A summary of the technologies and properties of the project:

- Optivision NACTM-3000 live streaming video servers located at each campus.
- VS-ProTM playback system at the University of Georgia School of Music.
- Streaming broadcast MPEG-2 video and dual channel audio.
- Musicians were simultaneously connected for the performance via timing tracks to a mixing board.
- Signals were merged into a final, recorded song.

The World's First Remote Barbershop Quartet. The goal was to orchestrate a multi-location barbershop quartet over Internet2 networks. Pieces played were "The Beer Barrel Polka", "In The Good Old Summertime", and "The Internet2 Song". Some of the setup details and lessons from the experiment include:

- The quartet was rehearsed via web.
- Each of the 4 singers in different cities, conductor in 5th city.
- Audience in 5th city along with mixer.
- Network delay variations prevented the singers from seeing or hearing each other and from seeing the conductor.
- Technical means were needed to deal with the network delays.

More information[17] can be found at the web site of the Society for the Preservation & Encouragement of Barber Shop Quartet Singing in America [84].

QoS Enabled Audio Teleportation. The goal of this project was to stream professional-quality audio to remote destinations using established Internet pathways. The setup involved the conference site in Dallas connected to CCRMA (Stanford) for the SuperComputing 2000 conference. A summary of the project's features:

- Real-time Internet transmission of CD-quality sound at 750 Kbps.

[16] http://www.umaine.edu/it/internet2/11600.html
[17] See also http://www.internet2.edu/presentations/20010308-I2MM-VidConfDevel&Deploy-Dixon.ppt

- TCP/IP streaming with QoS bounds on latency and jitter.
- Two-way telephone-style communication, streaming audio without buffering from a remote tape deck.
- Two musicians, from separate booths in Dallas, played "together" in the same space on the Stanford campus, but delay was severe.

[need reference]

2.5.2 Tele-immersion and data visualisation

Tele-immersion ***Office of the Future, CAVEs, collaboration within a virtual environment for the manipulation and visualisation of large amounts of data.***

[[[Need a paragraph on data visualisation.]]]

2.5.3 Remote control of scientific instruments

SOAR. enables the real-time control of such devices using reliable embedded execution from remote locations, allows the view of high resolution image data and utilises H.323 based videoconferencing enabling human supervision of the data acquisition process. [[[[Does this text belong to the description of SOAR? It reads like a misplaced generic description of telescope-control projects.]]]

The Southern Astrophysical Research (SOAR) Telescope project[18] is a 4.2-meter telescope funded by a partnership between the US National Optical Astronomy Observatories (NOAO), the country of Brazil, Michigan State University, and the University of North Carolina at Chapel Hill. The telescope will support high-quality imaging and spectroscopy in the optical and near-infrared wavelengths.

MAGIC. control data involve the adjustment of focus of the microscope, where high quality images of the microscope are also transferred over the high-speed network. The possibility of integrating the application into a tele-immersive environment is attractive as it will give a feeling of presence to the remote user. [[[As with SOAR, this seems like non-project-specific material that belongs elsewhere.]]]

The Microscope And Graphic Imaging Center (MAGIC)[19] provides local and remote access to optical and electron microscopes. Access is being provided to students and faculty at CSU Hayward and other educational institutions, including nearby community colleges. The mission of the Center is to expand the use of microscope imaging and analysis in science education and research. MAGIC is developing a model for remote access to scientific instruments[20]. This provides a way to share a variety of valuable resources with a worldwide audience. By pooling these resources and providing common network and user interfaces to them, science researchers and educators will have capabilities that no one institution could afford. Model software is being developed for interactive remote and shared access to an unmodified Philips XL 40 scanning electron microscope (SEM) located within the MAGIC facilities at CSU Hayward. A wide range of network technologies are being used to control the SEM, including modem, ISDN, Ethernet, T1, and ATM. A wide range of image transmission technologies are being used, including closed circuit TV and compressed video over ATM.

[18] http://www.soartelescope.org/
[19] http://www.csuhayward.edu/SCI/sem/
[20] http://www.csuhayward.edu/SCI/sem/remote.html

nanoManipulator The nanoManipulator [47] is a virtual-reality interface to scanned-probe microscopes. The nanoManipulator enables scientists to view a surface on a nanometer scale and use haptic devices to manipulate objects at this scale. What makes this application particularly interesting is the use of different computers to handle different modules of the system, such as the graphics, the haptics, and the microscope communicating over a high-speed Internet connection (this application is called the tele-nanoManipulator). The distributed users can use tele-immersion and audio and video links to facilitate collaboration.

2.5.4 Data Grid projects

GridPhyN. The Grid Physics Network *GridPhyN* project [46] is primarily focused on achieving IT advances in creating petascale virtual data grids (PVDGs). The project will package software and technologies to enable distributed collaborative exploration and experimental analysis of data, creating a multi-purpose, domain-independent Virtual Data Toolkit, and will use this toolkit to prototype PVDGs. The aim is to provide support to four frontier physics experiments that explore the fundamentals of nature and the universe. These experiments are:

- The CMS [108] and ATLAS [109] experiments, at the Large Hadron Collider at CERN, explore the fundamental forces of nature and the structure of the universe.
- The LIGO (Laser Interferometer Gravitational-wave Observatory) [1] will detect the gravitational waves of pulsars and other star systems.
- The SDSS (Sloan Digital Sky Survey) [113] will carry out a systematic sky survey enabling the study of stars, galaxies and other large structures.

The above experiments offer great challenges for data-intensive applications in terms of timeframes, data volumes and data types, and computational and transfer requirements.

DataGrid. *DataGrid* [29], a European-funded grid project, concentrates on several computation-intensive scientific projects:

- High Energy Physics (HEP), led by CERN.
- Biology and Medical Image processing, led by CNRS (France).
- Earth Observations (EO), led by the European Space Agency.

Chapter 3

Behaviour and QoS requirements of audio-visual applications

In this chapter we discuss the properties of advanced applications in terms of their end-to-end requirements for operation with high (or acceptable) quality. We also provide a review of research on the effects of network performance parameters, such as delay, loss, and jitter, on the qualitative behaviour of Internet applications. Because advanced applications are structured as ensembles of high quality data and media flows, their quality and usability can be assessed by (i) the quality of their individual flows and (ii) the degree to which requirements posed by interactions among the individual flows are satisfied. We follow this approach of studying the QoS requirements of advanced applications, that is, we examine the QoS properties of the individual, elementary flows and the requirements that arise from their relationships within the application scenario.

3.1 Introduction

3.1.1 What is application quality?

Network QoS allows the definition of quality metrics based on a variety of parameters. However, such QoS models are engineered using network-centric quality parameters (available bandwidth, delay, jitter). Application developers and users, on the other hand, require quality models that are geared to their needs, and that are expressed by different performance characteristics such as response time, predictability, and consistent perceptual quality. These are metrics that define what is called *application quality*. The term "application quality" is too vague to be deterministically defined. The reason is that the factors that influence quality are very "fuzzy". Such factors include the user's expectations and experience, the task of the application, and whether the application delivers the expected levels of performance. Furthermore, other factors, like charging for the use of the underlying network resources or the service, also influence application quality.

Another aspect of quality — one which is usually underestimated and overlooked — is the social behaviour of individual applications in an environment like the Internet. Applications that aggressively try to acquire as great a share of network resources as possible, in a network that does not impose any direct penalties for doing so, may seem to increase their own quality in comparison to applications that are more cautious (like those based on TCP transmission). While on a first look this does not seem to directly influence the quality or acceptability of an application, it may have undesirable implications for the health and well-being of the network. Unresponsive and excessive use of resources by applications can lead to severe congestion conditions

that will inevitably affect the irresponsible applications as well. So, the ability of an application to adapt to the state of the network may be considered an indirect aspect of its quality. While network operators and designers devise and deploy mechanisms in the network to help prevent such behaviour (directly, as with pricing or changes in queuing principles (RED), or indirectly, as with ECN), the cooperation of the applications is also required. Applications that can deal with changing network conditions and adapt (or adjust) their resource demands in response to indications of network performance variation are called *network-aware*. An application can be network-aware only if it is able to change its resource demands by appropriately changing its behaviour to work in different modes (adaptive applications, see section 2.1.3).

We therefore recognise some important features of application behaviour that are tightly coupled to the perception of their quality:

- Adaptivity. Applications should be designed with adaptivity in mind; they need to employ built-in mechanisms that allow them to probe the conditions of the network environment and alter their transmission characteristics accordingly. This means reducing their transmission requirements in periods of high demand (or congestion) and being able to quickly sense the availability of additional resources that can be used to increase their quality. In this way applications not only achieve an efficient utilisation of the resources that are available to them, thus maintaining a good balance between offered resources and quality, but also contribute to the well-being of concurrent applications and the stability of the network by fairly sharing common resources.

- Tolerance. Applications will face a wide range of network conditions and offered resources. These may appear in the form of variable bandwidth availability and delay, as well as loss of application data. Under these circumstances, applications should employ mechanisms that are capable of ameliorating the effects of such events. Due to the nature of the transmitted application data streams (e.g., audio or video data), there is an inherent amount of tolerance to varying network conditions, a range with which these conditions can vary without seriously affecting end-to-end quality. Tolerance to worsened network quality of service can be increased by introducing measures such as adaptive playout buffering and error correction and concealment.

3.1.2 Network QoS parameters

The most important metrics that characterise the performance of an IP network, and that are the most significant factors that influence the end-to-end quality of an application, are:

- *Bandwidth.* This signifies the portion of the available capacity of an end-to-end network path that is accessible to the application or data flow. Consequently, the number of bits that are injected into the network by the various flows of an application have to be adjusted accordingly.

- *Delay.* Network delay corresponds to the time it takes for application data units to be carried by the network to the destination. Network delay is caused by the combination of network propagation delay, processing delays and variable queuing delays at the intermediate routers on the path to the destination host [5]. A great deal of delay may cause data unavailability and unintelligible real-time interaction, with frustrating consequences for the application user.

- *Delay variation or jitter.* Delay variation is usually caused by the buffers built up on routers during periods of increased traffic, and less often by changes of routing due to failures or routing table updates. The amount of network jitter that an application can tolerate depends on a number of factors, the most important being the nature of the application tasks. It is also dependent on delay variation introduced by other systems on the application end-to-end path (like the operating system or the media encoding

process). Applications with stringent delay requirements can afford minimum delay variation. Other applications can make a trade-off by delaying presentation enough to build up a buffer that smooths out jitter. Still, there are limits to buffering; these limits depend on the delay tolerance of the application and on buffer memory constraints. Extensive jitter may require deep playback buffers that cause unacceptable delays before data are decoded, displayed or processed; this makes it difficult to auto-tune the depth of playback buffers.

- *Packet loss.* Packet loss is typically the result of excessive congestion in the network. Packet loss is defined as the fraction (or percentage) of IP data packets, out of the total number of transmitted packets, that are lost somewhere along the path from the source to the destination [5]. For most applications (especially multimedia applications) the packet loss percentage itself is not always enough to assess the impact on quality. In such cases the pattern of loss is also a significant parameter.

- The *loss pattern* or the *loss period* is often equally important, as it may offer valuable hints to the designers of codecs and applications. The most notable observation concerning loss patterns in today's Internet is that losses appear in bursts — usually short, but occasionally, at times of extreme congestion, more lengthy. Isolated losses tend to appear as well. Depending on the encoding mechanism used, the type of data being transported, and whether error correction or concealment techniques are being used, loss may cause either minor distortions or significant deterioration of quality. There has been a considerable amount of measurement and research aimed at characterizing and modelling packet loss patterns in the Internet [18, 19, 20, 88, 146].

3.1.3 Application QoS metrics

The network QoS metrics mentioned above do not necessarily coincide with the application's perception of QoS parameters. From the user's perspective, application performance is not concerned with the details of how a network service is implemented and how it performs. Application performance deterioration should be expressed in terms that focus on user-perceivable effects, rather than on their origin within the application end-to-end path. For example, the end-user of an application sees only one latency, and cannot distinguish the cause of it, whether it comes from the network or from the processing done on the end-system. Thus, from a user perspective, network QoS performance is hidden in end-to-end application-level performance.

Application-level performance metrics include:

- *Throughput.* This is the effective share of bandwidth that the application is getting from the network.

- *Latency.* This is the end-to-end delay that the application experiences. Delay has a direct impact on user satisfaction. The application does not determine its latency requirements only quantitatively but also qualitatively. For example, "the end-to-end requirement is such that intelligible and non-overlapping conversation is achieved" or "the amount of wait time is below some threshold" or "short response times is expected".

- *Availability and continuity of service.* This describes the requirement for uninterrupted service with acceptable quality. There are several factors that may disrupt the continuity of the service; among them, the most important are:

  - *Information (or data) loss.* At the user-perception level, information loss does not necessarily coincide with data packet loss at the network level. It might be data loss at the application level (for example, loss of signal fidelity due to encoding), or it might be completely user-specific. In this case, it is translated as a "user perceivable" loss of information that leads to user discomfort, lack of timely

judgement and reaction needed to execute the task successfully, annoyance, disorientation, lack of interest, etc. Therefore, information loss corresponds to the amount of information (visual content as the user sees it, loss of audio clarity, etc.) that a user perceives as missing. Note that this metric may have a subjective component as well. For example, a user might believe that something is wrong with a stream even in a lossless network environment. Also, more commonly, despite the fact that data are lost on the network, a user might not understand or perceive this (for example, if packet loss affected a non-audible, non-visible, or non-perceivable part of the transmitted information). This is an extremely interesting observation that can contradict the "bounded" approaches to QoS requirements and "ranges of tolerance".

– *Delay variation.* While certain levels of jitter can be absorbed by the application through the use of buffering, excessive variation of end-to-end delay means that application data packets may reach the receiver too late to be useful. Delay variation can also produce temporal inconsistency in presenting the data to the user (e.g., video freezing and then starting again) which results in the user becoming irritated and losing the sense of presence. In time-critical operations as well as human-to-human interactions, this is worse than having a constant delay. For services intolerant to delay variation, jitter can be removed by means of application buffering, but this comes at the expense of adding an additional delay component to the end-to-end latency.

– *Denial of Service (DoS) and security.* DoS causes service unavailability despite the underlying transport infrastructure being fully capable of offering the requested QoS. Security is also considered an important application-level QoS feature, as it offers trusted communication services (private meetings, secure transactions, etc.).

It is very important to emphasize here that the above application QoS metrics are not only affected by the network-centric metrics. Several other factors in the end-to-end application path may result in undesired changes of performance parameters: the operating system's inability to support the application, erroneous application and protocol-stack implementations, the usage environment (e.g., faulty equipment), etc. In most of these cases, when people investigate how application quality or utility is affected, it is the application-level and not the network-level performance parameters that they are able to measure, understand the influence of, and make recommendations about the desired values or properties of. This should be highlighted to avoid erroneous conclusions and misconceptions from network designers and researchers. For example, it is known through user trials and experiments that for videoconferencing applications interactivity becomes problematic when one-way delay is higher than 300–400 ms. This does not mean that the requirement of the application from the underlying network is just for a transmission delay below 300–400 ms. Other delays introduced along the end-to-end path (hardware capture device delays, encoding and buffering delays, scheduling delays from non-real-time operating systems) mean that the network delay requirement is even lower. In fact, it was demonstrated in [90] that in some cases during the transmission of high quality HDTV, the limitations of the end-system (PCI cards, network interface cards)[1], rather than the limitations of the network itself, were the main limiting factor. (Note however that the experiments described in [90] were conducted in conditions of a lightly-loaded network).

In the investigation of application QoS requirements that follows in this chapter, recommended values for performance parameters obtained though usability tests or other methods refer to the end-to-end application path. This should be taken into consideration in choosing appropriate network QoS values.

[1]In these experiments, non-specialised hardware was used to show the feasibility of displaying HDTV on commodity desktop PCs

3.2 Quality requirements of interactive voice

Voice over IP (VoIP) is characterised by stringent requirements regarding end-to-end delay, jitter and packet loss. While VoIP has a requirement for modest yet sustained bandwidth (64 Kbps and below are enough to accommodate a voice stream; use of more sophisticated encoding algorithms can reduce this to a few Kbps), bandwidth alone is not sufficient from a user QoS point of view. In this section, we discuss work that quantifies the effects of delay, delay variation and packet loss on interactive IP voice.

Voice quality is very subjective, and can be expressed primarily with respect to the individual user. At a high level, perceived IP voice quality is mainly affected by three factors:

- *Clarity.* This refers to the fidelity, clearness and intelligibility of the voice signal.
- ***End-to-end delay.***
- *Echo.* Echo is perceptible when the round-trip delay is above a certain threshold. It originates from the talker's voice "leaking" from the transmit path to the receiving path, that is, from a non-ideal separation of the send and receive signals. In a purely digital domain, this is never a problem, but end-to-end VoIP is seldom purely digital. Echo may originate from feedback in handsets or from PSTN gateways in the path. When the time difference between the original voice and the echo is short (25–30 ms), then the echo does not cause any disruption to the conversation. However, as Internet delays often exceed the 30 ms threshold, echo cancellation is typically used.

While the above factors originate from different sources, the individual contribution to disruption they cause cannot be distinguished by the human user of the application. It is difficult to categorise the effects that each of the above individually has on quality; only their joint effect can ultimately be measured. In the following, we study what qualitative effects the end-to-end values of delay, jitter and information loss have on IP voice, and suggest approximate desired values for these metrics.

Table 3.1: Delay guidelines for VoIP

One-way delay	**Effect on perceived quality**
< 100–150 ms	Delay not detectable
150–250 ms	Still acceptable quality, but a slight delay or hesitation is noticeable
Over 250–300 ms	Unacceptable delay; normal conversation impossible

Table 3.2: Jitter guidelines for VoIP

Delay variation (jitter)	**Effect on perceived quality**
< 40 ms	Jitter not detectable
40–75 ms	Good quality, but occasional delay or jumble noticeable
Over 75 ms	Unacceptable; too much jumble in the conversation

3.2.1 Effect of delay

Delay is introduced at all stages of a VoIP system: the sound-capture device, the encoding and packetisation modules, network transmission, receiver buffering, and decoding and playout of the signal. However, a user does not care where latency has been introduced; the user's requirement is to have low latency in order to

experience a truly interactive feel to inter-personal communication. Delay mostly affects conversational quality rather than received voice fidelity. There is a range of opinions on suitable ranges of one-way delay. According to ITU [63], users cannot notice any delay below 100–150 ms. Delay between 150 and 300 ms is perceived as a slight hesitation in the response of the conversational partner. Delay above 300 ms is obvious to the users; the conversation may be almost impossible, as each speaker backs further and further off to prevent interruptions. Talker overlap (the problem of callers stepping on one another's speech) becomes significant if the one-way delay becomes greater than 250 ms. ITU-TG.114 [63] recommends 150 ms as the maximum desired one-way latency to achieve high-quality voice. Vegesna [128] suggests a target of 100 ms end-to-end one-way delay in order to maintain the interactive nature of communication, while Kumar [72] argues that in order to maintain full-duplex voice conversation a one-way delay of less than 300 ms is desirable. Table 3.1 stands as a rule of thumb for the desired ranges of one-way delay.

3.2.2 Effect of jitter

Jitter results in variation in inter-packet arrival times. A jitter buffer can be used to alleviate the jitter effect; incoming packets are buffered and then read out at a nominal rate. Jitter buffer sizes usually adapt to instantaneous network jitter conditions. However, packets arriving very late are either discarded and considered lost (thus causing conversational gaps) or they obstruct the proper reconstruction of voice packets (generating a confusing conversation in which the talking parties may jumble together). Table 3.2 gives an indication of the perceptual effect of jitter.

3.2.3 Effect of packet loss

The study of the impact that the loss of data has on perceived quality is rather difficult, as it relies on a number of factors: the audio codec used, the existence of error protection or correction (e.g., FEC), and the pattern of packet loss itself. For example, packet loss events in which packets are lost individually are less harmful than loss events that occur in bursts with higher lengths. This is because packet repair techniques may be employed to recover from isolated lost packets, but may be unable to recover from a lengthy series of consecutive lost packets. Furthermore, the effect of loss on perceived quality depends on the location of the loss within the bitstream. Loss at an unvoiced segment has little impact on perceived quality, but this is not the case if a voiced segment is affected [112]. The effect of packet loss also depends on the packet size. When small packets (20 ms worth of audio) are used, the impact of a lost packet might easily be alleviated by simple error-concealment techniques at the receiver. However, this is more difficult to achieve when larger packets are transmitted (80 ms / 160 ms)[2]. Table 3.3 shows the effect of packet loss on voice for some of the most commonly used voice codecs. As the table shows, in most cases acceptable (toll) voice quality can be achieved if one-way delay is kept below 150 ms and packet loss below 2%. Resilience to packet loss can increase with the use of error correction and concealment techniques (FEC, interleaving, retransmission, etc.; see [91] for a survey), but as a side effect, these techniques increase end-to-end delay.

Figure 3.1 shows the distortion (R values), of different speech codecs for transmission under various one-way delay and packet loss ratio conditions. An R-value is the value calculated by the E-model, an objective speech quality assessment model (see section 4.2.2.6).

Apart from the above factors, further impairments can be caused by the specific selection of audio codec. These impairments are due to the distortion introduced by the codec and to the interaction of network effects and the codec. Table 3.4 summarises the most commonly used speech codecs with respect to three major factors

[2]On the other hand, small packet sizes increase the packet header overhead (RTP, UDP/TCP, IP) incurred. The selection of the audio packet size is a trade-off involving the specific audio codec, the impact of a lost packet, and the packet header overhead.

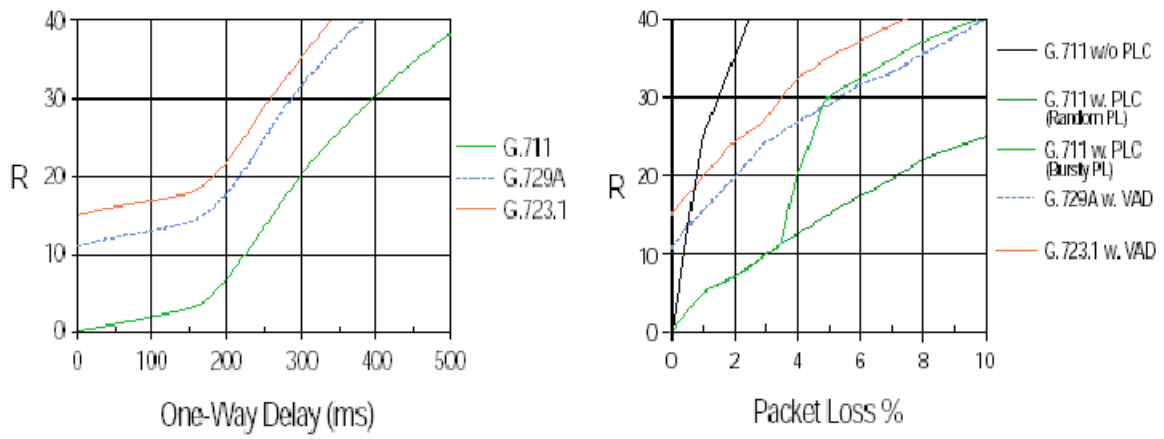

Figure 3.1: Effect of one-way delay and packet loss on voice distortion for various voice codecs. Perceived distortion measured using R-ratings, as $100 - R$. (Source: [85])

Table 3.3: Effect of packet loss on voice quality
(source: AHM Technology, `http://www.ahmtech.com/`)

Codec	**Latency(ms)**	**Packet Loss(%)**	**MOS**
G.711 w/o PLC	150	1	3.55
G.711 w PLC	150	1	4.31
G.711 w/o PLC	150	2	3.05
G.711 w PLC	150	2	4.26
G.729A+VAD	150	1	3.99
G.729A+VAD	150	2	3.82
G.723.1A+VAD	150	1	3.82
G.723.1A+VAD	150	2	3.60
G.711 w PLC	400	0	3.60

that impact speech coding: bit rate, quality and complexity.

As a general remark, in terms of network QoS parameters, bandwidth is not the main issue for voice traffic. The requirements of this class of applications is that the other network parameters, like end-to-end delay, jitter and loss, be kept within well-defined boundaries.

Experience from the tremendous acceptance of mobile or cellular telephony[3] has proven that human users are prepared to tolerate transient bad voice quality when communicating, given that the appropriate incentives are in place (e.g., it's the only means of achieving voice communication, or it's considerably more economical than the available alternatives). Nevertheless, it needs to be stressed that the context of use (criticality of application, quality expectations) is still the most influential factor.

[3]The encoding quality of the voice signal is usually modest, and can be low at times, depending on location and signal strength. Furthermore, fluctuations in quality, temporary loss of service, and complete loss of service are also quite common.

Table 3.4: Speech coding standards

Codec	**Description**	**Rate (Kbps)**	**Quality (MOS)**	**Complexity**
ITU-T G.711	Pulse Code Modulation for voice frequencies (PCM); 4 kHz bandwidth	64	>4	—
ITU-T G.722	SB-ADPCM: Sub-Band Adaptive Differential Pulse Code Modulation; 16 kHz sampling frequency	48–64	3.8	low
ITU-T G.723.1	Dual rate speech codec for multimedia applications (MP-MLQ/ACELP)	6.4/5.3	3.9	high
ITU-T G.726	Adaptive Differential Pulse Code Modulation (ADPCM)	32	3.8	low
ITU-T G.728	4 kHz bandwidth; Low Delay CELP (LD-CELP) G.727H: Variable-Rate LD-CELP	16	3.6	low
ITU-T G.729(A/B)	Conjugate Structure Algebraic CELP (CS-ACELP) G.729A: Reduced complexity algorithm; G.729B: Discontinuous Transmission (DTX)	8	3.9	medium
ETSI GSM 06.10 (GSM FR)	Full Rate (FR) speech codec (RPE-LTP: Regular Pulse Excitation - Long Term Prediction)	13	3.5	low
ETSI GSM 06.20 (GSM HR)	Half Rate (HR) speech codec (VSELP: Vector Sum Excited Linear Prediction)	5.6	3.5	high
ETSI GSM 06.60 (GSM EFR)	Enhanced Full Rate (EFR) speech codec (ACELP: Algebraic CELP)	12.2	>4	high
ETSI GSM 06.70 (GSM AMR)	ETSI Adaptive Multi-Rate (AMR) speech codec	4.8–12.2	>4	high
Nokia AMR-WB	Nokia proposal for a wideband Adaptive Multi-Rate (AMR) codec	12.6–23.85	>4	very high

3.2.4 Additional sources of information on interactive voice applications and VoIP

The list below provides links and references to useful additional information not directly cited in the text.

- Several VoIP-relevant documents at Telchemy: http://www.telchemy.com/techref.html
- Solving QoS in VoIP: http://www.brooktrout.com/news/in_the_news/0701_solve_qos.pdf
- Measuring voice quality over VoIP networks: http://www.tolly.com/News/NewsDesk/20011219VSVoIP.asp
- Agilent Voice Quality Tester (VQT): http://www.onenetworks.com/vqt/J1981A_VQT_Features.asp
- TIPHON-Questionnaire: User acceptance of Voice over IP (VoIP): http://docbox.etsi.org/tech-org/tiphon/Document/tiphon/03-permanent/(98)02r2.doc

3.3 Quality requirements of audio transmission

The term "audio" refers here to any kind of sound signal that is not explicitly "voiced", such as music. We distinguish "voice" from "audio" because there are several important differences between them. In general, audio signals require higher bandwidth[4]. Thus a higher transmission bit rate is needed in principle. Depending on the application task (whether it is interactive or not), delay constraints can be either tight or relaxed. Delay, jitter and packet loss compensation techniques can be used, and we shall see when and how this can be achieved. As a rule of thumb, we can say that, in contrast to expectations for voice, humans will always expect high-fidelity audio, and will not tolerate quality degradations. This seems to be an outcome of users' expectations. For music to be acceptable, it has to be at least of FM-quality, and preferably CD-quality or higher, as this is what users are used to.

3.4 QoS requirements of digital video transmission

The transmission of video offers the opportunity for far more advanced forms of communication, collaboration and entertainment, but it also poses new and considerably higher demands for network resources when compared to other media. As an indication of the diversity of bandwidth requirements of video, the figures shown in Table **??** show the range of bit-rates for some popular video formats.

Table 3.5: Typical bandwidth requirements for some commonly used video formats

Video format	**Typical bandwidth requirement**
Uncompressed HDTV	1.5 Gbits/sec
HDTV, Interim format	360 Mbits/sec
Standard Definition TV (SDTV), SMPTE	270 Mbits/sec
Compressed MPEG-2 4:2:2	25–60 Mbits/sec
Broadcast quality HDTV (MPEG-2)	19.4 Mbits/sec
MPEG-2 Standard Definition TV (SDTV)	6 Mbits/sec
MPEG-1	1.5 Mbits/sec
MPEG-4	5 Kbits/sec–4 Mbits/sec
H.323 (H.263)	28 Kbits/sec–1 Mbits/sec

The network QoS requirements of a video flow are primarily determined by its application context. Depending on whether the application is telepresence or teledata and the degree of interactivity it involves, different handling from the network is expected, as discussed in section 2. In this respect, we identify two major modalities of networked video applications:

3.4.1 Interactive video

Interactive video is used in telepresence (e.g., videoconferencing and virtual collaborative environments), distance learning, medical applications (remote surgery), and scientific applications (immersive data exploration, remote control of scientific instruments). For these applications, short latency is an essential requirement if acceptable interactivity is to be maintained. Depending on the usage scenario (technology and codec used) and user expectations, throughput requirements range from a few hundred Kbps (N x 128 Kbps — H.261, H.263) for videoconferencing (384–800 Kbps per flow is currently used in typical H.323 conferences), to 2–10

[4]Recall that speech signals can be restricted to a 4 kHz bandwidth with almost no loss

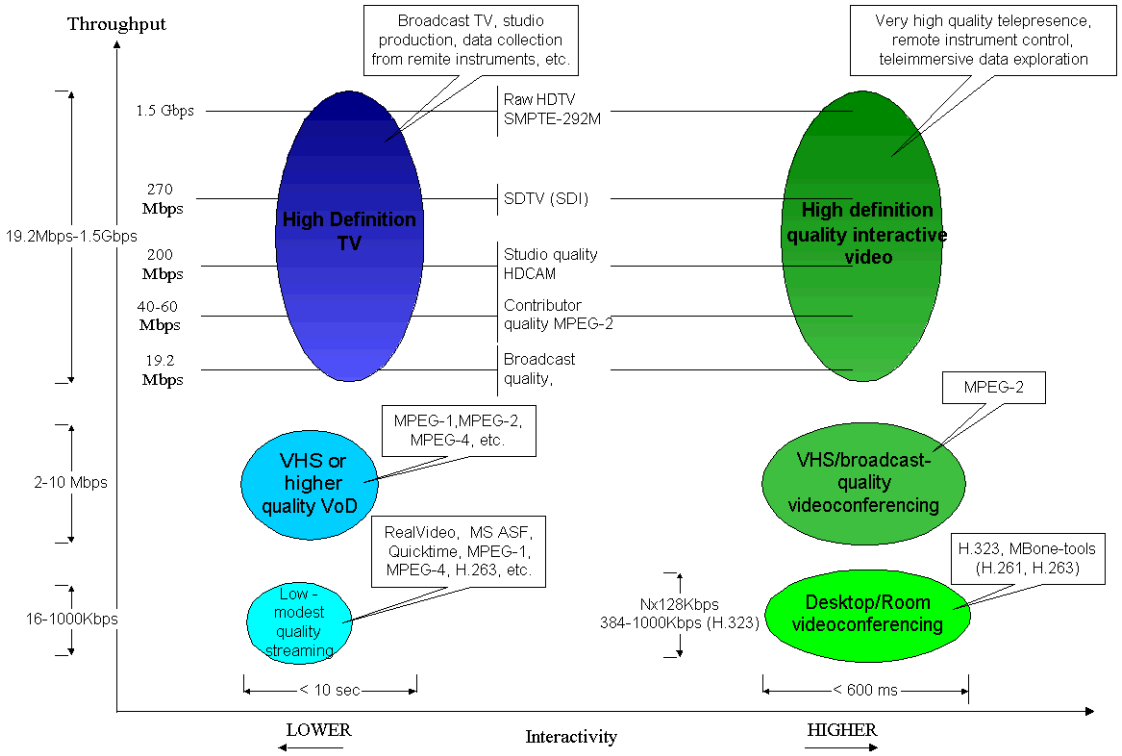

Figure 3.2: Throughput and interactivity requirements for common video services. On the right side of the plot are delay-sensitive interactive video services that may accept slightly higher packet loss rates. On the left side are services that can withstand larger latencies but are less tolerant to packet loss (teledata). Colour of higher intensity illustrates higher requirements for throughput.

Mbps for MPEG-2 based videoconferencing, to 19.2 Mbps–1.5 Gbps for HDTV-quality telepresence. Figure 3.2 presents typical throughput requirements for common interactive video services (shown in green to indicate their requirements for short latencies).

Two-way interactivity (human-to-human or human-to-machine) of these applications means that latency requirements are quite stringent. Some general rules of thumb for the requirements of interactive video apply:

- Interactivity means low end-to-end delay.
- Users can tolerate a certain amount of video distortion in tele-collaboration, which means that
  1. packet-loss tolerance can be higher (compared to streaming video scenarios) with acceptable quality.
  2. throughput requirements can be more relaxed, and rate adaptation can be used to reduce bandwidth requirements.
- Synchronisation between the audio and video flows (lip-synchronisation) is essential to maintain good intelligibility; thus, besides low delay, delay variation should be minimised.
- Short, bounded delay variation seems to be a main requirement for remote control operations as well, so that user actions correspond to events shown on the video feed of the remote location. (Figure 3.3)

H.323 videoconferencing. In section 3.2.1 we discussed delay requirements for two-way speech, depicted in Table 3.1. In principle, we would expect that these requirements would apply to interactive video (videoconferencing) as well. However, a major obstacle in achieving this is the encoding delay introduced by compression. It is reported in [104] that in current H.323 codecs the encoding/decoding delay is approximately 240 ms. In addition, each MCU in an H.323 configuration may generate additional delay of 120–200 ms.

Figures for acceptable data loss rates depend on the application technology. H.323 video is very sensitive to both jitter and packet loss. While some IP audio/video applications can withstand significantly high loss rates (even up to 10–15%) by employing redundancy techniques, to the best of our knowledge H.323 implementations do not currently use these techniques.

Audio-video synchronisation poses an extra requirement for timely delivery of all involved data flows; we examine this issue in more detail in section 3.4.3.4 below.

3.4.2 Video streaming

Video streaming refers to the real-time transport of live or stored video, such as news retrieval, live video transmission or multimedia information broadcasting (webcasting), Pay-TV, video on demand, network-based studio production and HDTV[5]. In video streaming applications interactivity is not the main feature, with the exception of VCR-like control functions (pause, stop, etc.). User expectations and application usage scenarios vary widely in terms of desired quality, and so do throughput requirements (Figure 3.2).

- Streaming of modest-quality recorded or live content (RealVideo, Microsoft Advanced Streaming Format (MS ASF), Quicktime) currently involves low expectations of throughput (a few tens of Kbps to 1 Mbps).
- Video on Demand (VoD) with equal or better than VHS quality. MPEG-1 and MPEG-2 have been the codecs of choice. The recent evolution of the MPEG-4 standard offers similar quality at lower bit-rates, and also the opportunity for the creation and manipulation of new types of presentations (synthetic video,

[5]We here ignore the case of applications that download the video material prior to its display (non-streaming video).

animation, integration of natural and synthetic scenes, etc.) offering a plethora of new ways of handling visual content (object-based encoding, content-based scalability). Throughput requirements range from 1–3 Mbps (MPEG-1) up to 10 Mbps or higher (MPEG-2).

- HDTV can be used for extremely high quality video for studio production, high-quality broadcast TV, etc. Depending on the compression used, throughput requirements range from 19.2 Mbps to 1.5 Gbps for raw HDTV (see section 2.4.2.2).

The main requirement of users of non-interactive streaming video is for sustained high visual quality. Given that an initial delay can be used to build a receiver playout buffer that can accommodate quite high jitter values, this translates into bounded data loss rates; as shown in Figure 3.3, the user expectation of data loss is that it should be lower than 2–3%. Today's video streaming is confined to modest-quality video with restricted resolution (image size). In such scenarios the distortions introduced by encoding with comparatively low bit-rates and packet loss are usually tolerated by users. However, advanced video services over IP use much higher resolutions where encoding distortions or lost information are easily spotted. Therefore, they need to maintain sustained high bandwidth and very low packet loss in order to meet the expectations of users who have experience with similar, non-IP, high-quality video services (digital broadcast TV, satellite, DVD and VCR). As the data loss tolerance is much lower in such services, packet loss needs to be kept to a minimum. Delay requirements are relaxed; there is usually minimal interactivity involved, so the application can afford a startup delay on the order of seconds (up to 10 seconds for video streaming, or even higher for VoD). Such an initial delay can be used to build up receiver buffers that improve the application's resistance to any variation in packet inter-arrival times and also to transient congestion. Transmission delay needs to be kept controlled in order to constrain jitter, but most likely larger delays can be tolerated in the case of streamed video than in the interactive case. Obviously users will expect relatively fast reaction in response to control actions (*start*, *stop*, *pause*, etc.), but such responses can take on the order of one second rather than on the order of tens of milliseconds. Such relaxed requirements also apply to delay variation where higher values of jitter, even up to 500 ms, are considered acceptable [123]. Efficient transmission-scheduling algorithms can be employed to maximise network utilisation and application quality. Furthermore, advanced packet loss protection mechanisms can be used to reduce the effects of information loss that results from transmission errors and packet loss.

For both of these types of video transmission, the bandwidth requirements are determined by the type of application and the capacity or nature of the network (Virtual Private Networks (VPN) or enterprise networks, public Internet, DSL, cable, wireless networks or telephone lines). Demand for bandwidth can span the whole range from very low bit-rates (a few Kbps) to extremely high transmission rates (e.g., uncompressed HDTV) as shown above. The video quality at the receiving end is a complex function of several factors: the channel rate availability (this determines the encoding quality), the frame rate, the resolution of the video image, etc. The end-to-end latency and delay variation influence the application's interactivity and the timely delivery of video data packets for decoding, and packet loss causes significant deterioration of the visual quality. The two forms of video tasks mentioned above have different requirements for delay and data loss, and we address these issues in the following section.

3.4.3 Effect of network transmission on digital video quality

The task of specifying the effects of network QoS parameters on video quality is difficult but very interesting. Bandwidth variability and increased delay, jitter and packet loss lessen the perceived quality or fidelity of the received video content. As a consequence, they affect the task of the application. In streaming applications, for example, high levels of distortion and quality fluctuations are very noticeable to the human observer, and may cause misunderstanding of a video clip or loss of content intelligibility. In collaborative telepresence applications

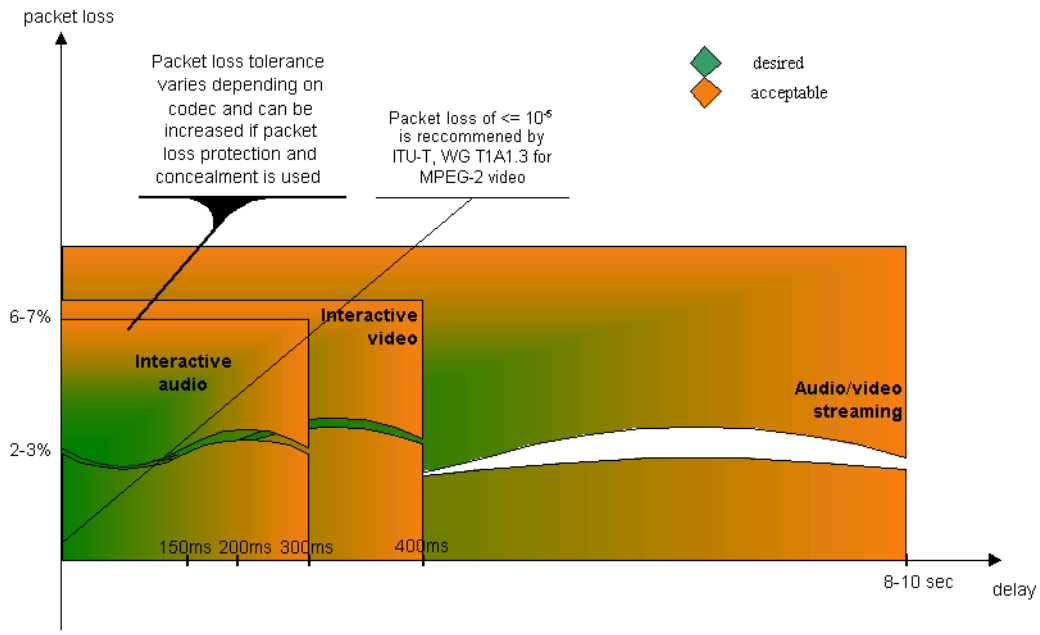

Figure 3.3: Mapping of user-centric requirements for one-way delay and packet loss for audio and video streams. The lower parts of the surfaces depict a user-centric expectation of these performance parameters. The upper parts show how tolerance of delay and loss can be increased, while maintaining user-acceptable quality, by employing packet-loss protection techniques ***(incomplete...).***

they result in loss of inter-stream synchronisation (e.g., between audio and video) and difficulties, or even breaks, in communication and coordination. The task of remote control operations can be severely affected, as the user is responding to video imagery that is out of date due to high latency or jitter. In the following section, we document the effects that the end-to-end QoS parameters may have on interactive or streaming video. Although we categorise these effects for each of these parameters individually, it should be noted that they do not affect quality in an independent manner. They act in combination or cumulatively, and ultimately, only this joint effect is detected by the application user. However, studying the effects of network parameters in isolation is a more tractable approach. The challenging issue for applications is to investigate what adaptation decisions and tradeoffs can be employed to reduce this cumulative effect on quality. As an example, in a lossy environment part of the bit-rate available to the video flow may by used to transmit redundant information to recover from lost packets.

3.4.3.1 Transmission bit-rate

High-quality digital video generates bandwidth-thirsty flows. Consequently, the throughput that a video stream receives predominates in determining its visual quality. We show the relationship between quality and throughput by examining Figure 3.4 which shows the evolution of perceived video quality as a function of the encoding bit-rate. The quality scores were obtained using an implementation of the ITS objective video quality metric [144] (see section 4.1.2.3). Objective video quality assessment metrics are computational models that measure the quality of a video sequence in a manner that produces results similar to those obtained by human observers (for more details, refer to section 4.1.2). The quality scores are in the 1 to 5 range. Three 150-frame-long CIF-size (352x288) video sequences were encoded using several bit-rates, ranging from 256 Kbps to 3 Mbps.

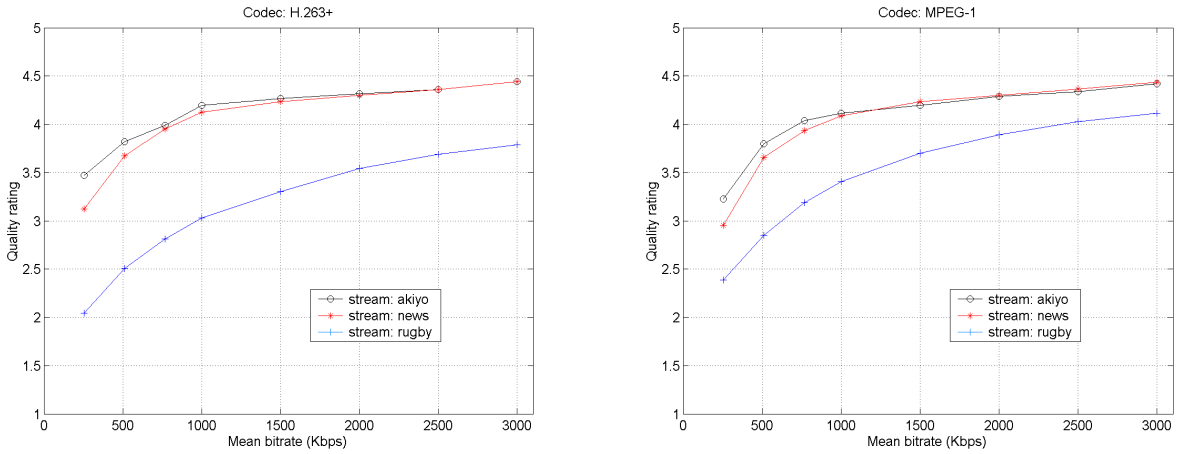

Figure 3.4: Video quality versus encoding bit rate for three 150-frame-long sequences (*akiyo*, *news* and *rugby*; resolution: 352x288) and two different codecs, H.263 (left) and MPEG-1 (right). Because the *rugby* sequence contains more motion and spatial detail, its quality for equivalent bit-rates is lower in comparison to the other two sequences.

The video quality is a convex increasing function of the encoding bit-rate. There is an initial range of bandwidth values where the video quality increases substantially. After that, the quality line saturates; any extra bit-rate offered to the encoder gives only marginal quality gains and is probably an unwise use of bandwidth. Knowledge of the point where the quality graph starts to flatten would be extremely important in determining the throughput requirements for controlled high-quality video. However, this is not straightforward, as several other factors influence the shape and the slope of the line; the complexity of the visual content and the video codec used are the most important ones. The visual complexity represents the amount of spatial detail and motion in the video sequence. This is also shown in Figure 3.4 where the quality is plotted for three video sequences: the video sequences *akiyo* and *news* have relatively little motion, where the sequence *rugby* has much greater spatial detail and significant motion. The *rugby* sequence lies below the other two, which suggests that more perceived distortion is present at equivalent bit-rates. This shows that the throughput requirement for a video application is highly dynamic, and thus the bandwidth offered by the network needs to be readjusted during the transmission according to the content's variability. The effect of the particular choice of video codec on quality is also shown in Figure 3.4; the different shapes of the two sets of quality plots correspond to the two video encoders used (H.263 on the left and MPEG-1 on the right).

3.4.3.2 End-to-end latency and delay variation

3.4.3.3 Packet Loss

The effect of packet loss on compressed video streams is particularly disruptive because the distortions packet loss causes to the image are typically more annoying to the human viewer than other types of impairments (e.g., encoding artifacts). The reason loss is so detrimental to video flows is that image compression achieves reduction in the number of bits required to transmit the stream by removing redundancies inherent in video data. Removing most of the redundancies by means of compression implies that any loss of compressed data cannot be recovered. The effects of packet loss depend on a number of factors, and the most important are: (i) the compression technique used, and the selection of encoding parameters such as the compression ratio, (ii) the data loss rate, (iii) the pattern of loss, and (iv) the data packet size. In order to reduce spatio-temporal

redundancies and achieve better compression ratios, most contemporary video codecs use quantisation and inter-frame compression by exploiting motion compensation and estimation. Inter-frame compression means that certain parts of the compressed bitstream, like those that belong to reference frames (I- and P-frames in MPEG-1 and MPEG-2 or intra-pictures in H.263), are more important than those that belong to predicted frames (B-frames in MPEG) as they are used as predictors for the decoding of predicted frames. Packet loss in a predicted or difference frame will only affect the particular frame, but packet loss in a reference frame propagates to all the dependent predicted frames, causing more severe distortion that lasts for the duration of the dependent frames. This phenomenon is called *propagation of errors*. The pattern of loss may also have an impact.The impact of a lost packet is higher if the packet is larger. Smaller packets, if lost, cause less distortion, but on the other hand, small packet sizes mean that more packets have to be transmitted, which increases the header and packetisation overhead and decreases the throughput.

As far as packet loss is concerned, the nature of the video application, whether we talk about interactive or one-way video streaming, does not seem to have any particular significance in terms of the effect on the visual content. Nevertheless, users are more tolerant of data loss in face-to-face video interaction (videoconferencing). Different types of video application tasks have completely different opportunities to recover from losses. Non-interactive applications have an advantage — they can afford longer latencies and consequently they can use more sophisticated error-recovery techniques. As already mentioned, the use of source FEC or retransmission introduces extra latencies that delay-sensitive flows cannot always withstand, while one-way streaming applications can employ this technique, as latency is not a crucial factor.

It is useful to have a measure of how IP packet loss affects video quality, but it is very hard to determine the degree of damage done by packet loss. The kinds of data packets that are lost is a crucial factor, as some carry more important information (syntax, motion vectors, blocks that carry more visual information, etc.) than others. Boyce et. al [21] define a frame error state measure, which is determined by whether or not a lost packet affects the frame. Although this measure does not give much of a clue about the perceived quality effect, it does show how packet loss affects frame integrity. The researchers found that even a small packet loss rate results in a higher frame error rate. For example, using real transmissions of MPEG-1 video, they observed that a 3% packet loss rate results in a frame error rate of 30%.

Particularly interesting is the interaction between throughput and packet loss. As shown in section 3.4.3.1, in the absence of packet loss, the quality of an encoded video stream is an increasing function of the available bit-rate. However, as reported in [129], when packet loss is present, quality increases up to a certain bit-rate, but then starts to smoothly decrease (shown in Figure 3.5). This can be explained: at lower bit-rates, the visual distortion caused by encoding artifacts is dominant, so quality increases with bit-rate. The perceived quality peaks and then starts to drop because the higher the average bit-rate the greater the number of packets being lost which, in turn, causes larger visual distortions. So there is an optimal average bit-rate, which is somewhat independent of the Packet Loss Rate (PLR) and directly dependent on the sequence (content) type [129]. This result shows that, in the presence of packet loss, increasing the bit-rate does indeed worsen the quality. This clearly shows the benefits of designing network-friendly applications that adapt themselves in response to congestion signals.

Several techniques can be used to alleviate the effect of packet loss. These include FEC and retransmission at the sender, and error concealment techniques at the receiver/decoder [134]. Some other researchers have proposed the use of more robust coding techniques for error-prone environments such as the best-effort Internet [76]. In order to avoid the persistence of error blocks throughout several successive frames, such techniques include the use of intra-frame encoding only, or they use frequent intra-updates of the frame blocks (macroblocks), called intra-refresh. However, nothing comes free; these techniques inevitably require more bandwidth for equivalent

[5]Such techniques are widely used today by most popular Internet video streaming applications, like RealPlayer and Windows Media Player

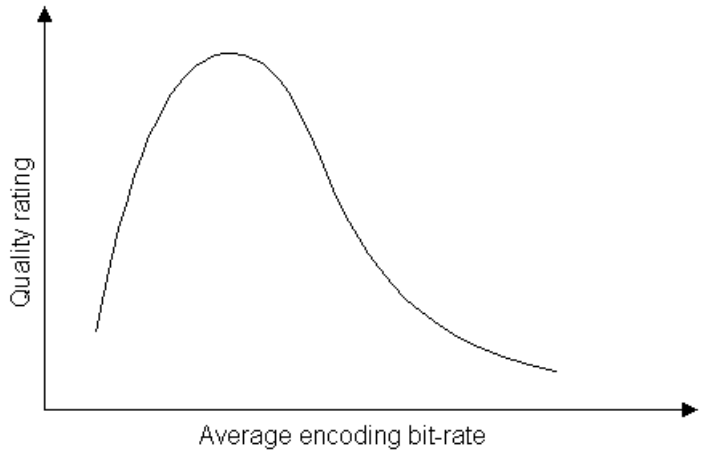

Figure 3.5: Video quality versus encoding bit-rate under packet loss conditions (constant PLR) for an MPEG-encoded sequence. We observe that the quality is increasing up to a certain bit-rate but then drops as the effect of lost packets increases with higher bit-rates.

encoding quality.

Figure 3.3 depicts performance targets for delay and PLR for audio and video applications, based on user expectations of application performance. The graph is based on indications and recommendations from Study Group 12 of ITU-T [87] and the literature study on the perceived quality of generic applications from the TF-NGN working group within the GÈANT project[6]. While the suggested tolerance for data loss is below 1% for high-quality audio-video streaming and below 2–3% for two-way interactive audiovisual services[7], the graph shows that the tolerance of audio and video flows can be increased by the use of FEC, e.g., [11]. Such models of user performance expectations, based on the user's perception of quality, "...provide with some upper and lower boundaries for applications to be perceived acceptable to the user and show how the underlying impairments of information loss and delay can be grouped appropriately..." [87]. [[[Check wording of this quote. Also, I found this graphic confusing.]]]

3.4.3.4 Interactions between the media in video-based services

If we want to achieve a natural, "in-sync" impression in multimedia presentations, it is essential to preserve the temporal relationship among several continuous media flows. Media flows that are "out-of-sync" are often perceived as artificial, awkward and annoying [110]. While many systems (e.g., MPEG) allow multiplexing of the various streams to avoid this phenomenon, this is not always possible, due to the nature of the participating flows, nor it is always desirable, as different media streams may be handled by different modules. Avoiding application-level multiplexing of the media streams can sometimes provide more opportunities to apply different transmission, adaptation and error-protection strategies to the individual flows. In such cases synchronisation is an additional requirement.

We observe two modes of synchronisation:

- Intra-stream synchronisation refers to the time relation between the various presentation units of a media

[6] http://www.dante.net/tf-ngn/

[7] As mentioned earlier in section 2.1.1, this is due to the higher tolerance of the human eye and ear for data loss.

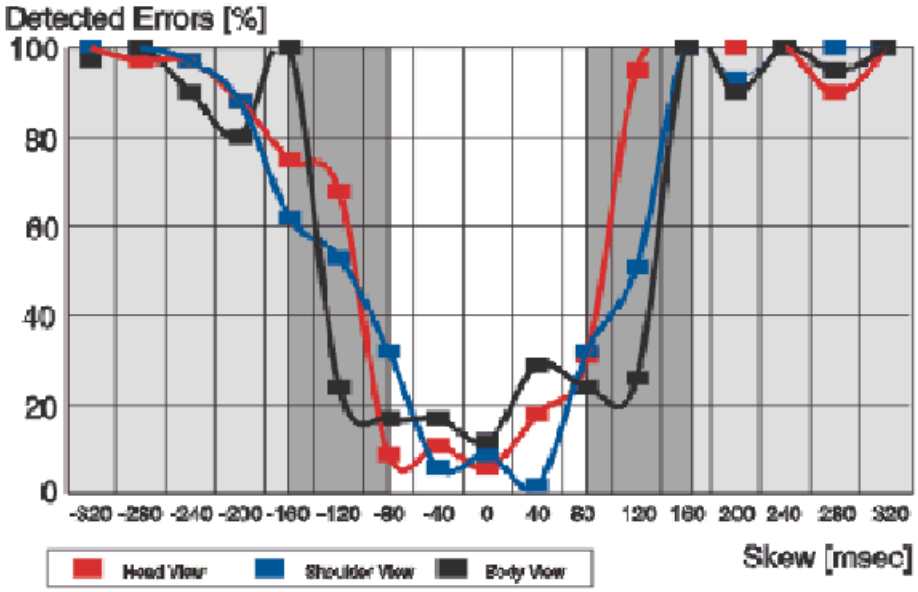

Figure 3.6: Detection of errors in lip-synchronisation for different values of skew (time-difference between audio and video). Also shown in different shading are areas related to the detection of synchronisation errors. (Source: [110], ©IEEE).

object. For example, for full-frame-rate video at 30 frames per second, the consecutive frames have to be available for presentation at approximately 33 ms intervals. Loss of intra-stream synchronisation is the result of undesired values of application QoS parameters. When frame-skipping is used to reduce bandwidth consumption, loss of synchronisation takes the form of a jerky presentation. Excessive jitter and packet loss may cause some media data units to be unavailable for timely presentation, thus causing gaps in the presentation. These effects have been examined in sections 3.4.3.2 and 3.4.3.3.

- Inter-stream synchronisation refers to the timing relationship that has to be achieved between the media streams in order to preserve presentation quality. As audio is a requirement for most Internet-video-based services, it is important to see how the network QoS parameters affect the interactions of audio and video. *Lip-synchronisation* refers to the temporal relationship between audio and video in the special case of human speech. This is known to be the most demanding type of media synchronisation in terms of preserving timing requirements. Loss of synchronisation occurs when there is a noticeable time difference between the related audio and video data units; this time difference in known as *skew*.

Let us examine inter-stream synchronisation, emphasizing lip-synchronisation issues. We try to map lip-synchronisation requirements to network QoS metrics by discussing and analysing human perception results[8] reported by Steinmetz in [110] and shown in Figure 3.6. The graphs in the figure depict the level of detection of synchronisation errors for different skew values. From the graph it can be seen that most humans do not detect any lag when the absolute skew value is 80 ms (-80 ms — audio behind video; +80 ms — audio ahead of

[8]Obtained by users viewing a video clip of a person talking. Three views of the person were used: head view (speaker very close), shoulder view, and body view (speaker relatively far away) [110].

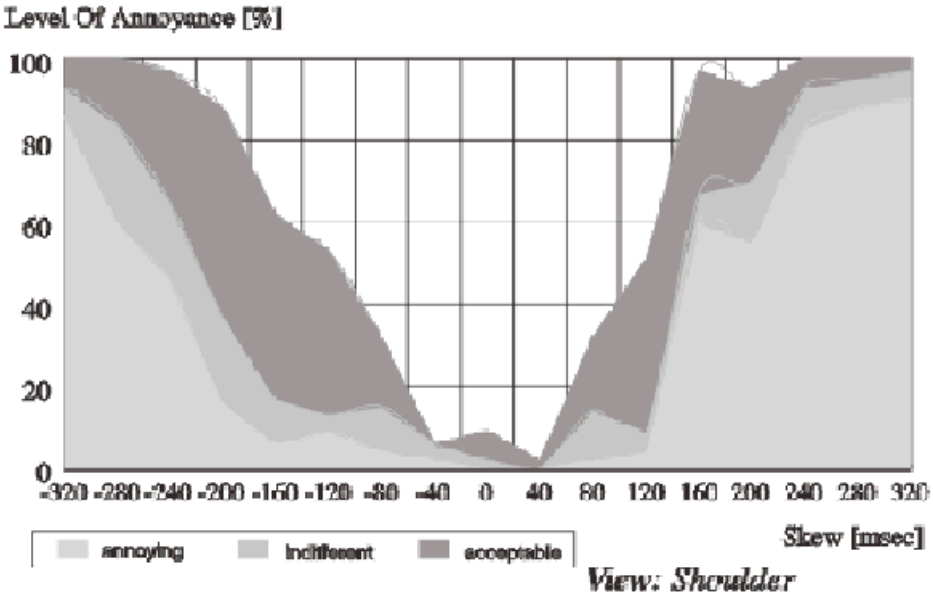

Figure 3.7: Level of annoyance of synchronisation errors for various values of skew. (Source: [110], © IEEE.)

video). In the areas where absolute skew spans over 160 ms, error in synchronisation was reported by almost all subjects; this caused the subjects to be distracted by the out-of-sync effect rather than attracted by the viewing content itself. In the transitional areas between 80 and 160 ms (absolute skew), detection of synchronisation errors depended on how close the speaker was — the closer the speaker, the easier the errors were detected. In the transitional areas, synchronisation errors were more easily detected when audio was ahead of video (steeper slopes of the detection error in the [+80, +160] ms skew region, gentler slopes in the [-80, -160] ms area). This means that video being ahead of audio could be tolerated better than the reverse. Experiments with other languages (Spanish, Italian, French, Swedish) showed similar behaviour. Results were also obtained for different video content (a violinist, a choir) and showed that they did not pose any requirements beyond those posed by the speakers.

Based on the results of perception of synchronisation errors shown in Figure 3.6, Figure 3.7 depicts how different levels of skew were qualified (acceptable, indifferent, or annoying) in the case of the shoulder view. (For a complete view of the result graphs refer to [110].) The upper (envelope) curve shows the proportion of subjects that detected any loss of synchronisation. The authors conclude that skews between -80 ms and +80 ms are acceptable to most viewers, and thus 80 ms lag between audio and video is an upper threshold for maintaining a synchronised audio-video presentation. Similar thresholds are also reported in [87].

Iai et. al [57] report the effects of video and audio delays on quality through a series of subjective tests on conversational video. These results are reproduced in Figure 3.8. The hatched region shows where quality is acceptable, and the heavy lines bound the regions where the inequality in speech and video delays is detectable. When speech delay is shorter than video delay, then the detection threshold is about 120 ms. When speech delay is longer than video delay, then the detection threshold is about 250 ms. The detection level for the delay difference was defined as the value at which the detection rate was 0.5; this is why the threshold values reported here are somewhat higher than those discussed above. But the finding that inequalities in speech and video delays are more easily detected when the speech precedes the video is in agreement with Steinmetz's

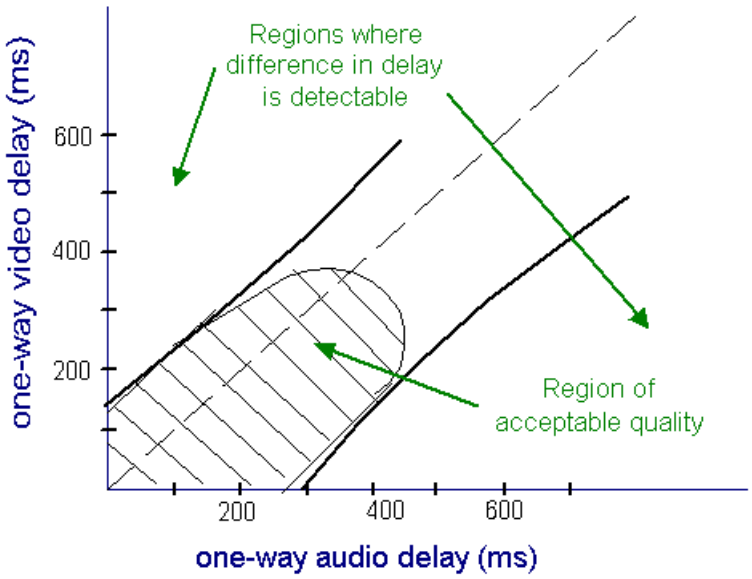

Figure 3.8: Acceptable region of end-to-end delay for the audio and video parts of a videoconferencing task. Also shown are the areas in which the difference between the audio and video delay is noticeable (loss of audio-video synchronisation).

results [110].

As a general conclusion, a skew of 80 ms between the audio and video flows in a multimedia presentation is the upper threshold to preserve synchronisation without noticeable effects. There are two sources of network distortions that may result in synchronisation errors: packet loss and jitter. Lost packets result in a presentation gap as the corresponding audio or video data units are not available in time. When a video frame is lost, then stream lag will occur. Delay variation between the two flows also results in a lag between the presentation of audio and video segments that should normally be displayed simultaneously. This problem can be partially solved by introducing a short initial delay to allow a de-jittering buffer to be built up. Different jitter for the two flows means that respective data units arrive at different times even though they were generated at the same time. If the corresponding audio and video segments are appropriately time-stamped upon generation or transmission, then at the receiving side they can be scheduled for later display in order to allow for both segments to be received. This assumes that the difference between the jitter of the two streams is below some threshold. Also, with interactive video, there is a limitation on the amount of initial buffering.

[*Study of synchronisation effects with other media types (e.g., remote instrument control) should be documented as well*]

3.5 An application-network cooperative approach to application QoS

This section discusses issues that promote the idea of cooperation between the application and the network in order to provide the necessary network stability and application quality. It highlights the idea that application designers should build applications that can be aware of the underlying network conditions and react to them (adaptation). It also promotes the necessity of congestion adaptation in conjunction with any optional offered

QoS from the network. We examine the question: What choices can the applications make in order to respond to varying network conditions? We briefly talk about TCP-friendly adaptation for continuous media flows and why it is important, and what adaptation mechanisms are available for Internet audio and video. We also ask: What are the proper design and run-time choices for video adaptation, and how are these choices influenced by the type of video task and the specific application requirements?

Recent attempts to implement network QoS within Internet2 have proven the difficulties of employing QoS models that offer hard QoS guarantees. Such techniques require considerable investment in designing, testing and planning, and demand for them is still weak. Development of such services, like the QBone Premium Service [34, 83] still lacks support from router vendors and, more importantly, requires dramatic network upgrades with considerable operational and economic costs. In light of such scalability problems, IP-layer differentiation could focus on "non-elevated" services that can be deployed incrementally and cheaply over the existing Internet. Such services divide today's monolithic best-effort service class into multiple "different but equal" service classes that trade off delay and loss. Such approaches include Alternative Best Effort (ABE) [56] and Best Effort Differentiated Service (BEDS) [36]. While these services will obviously require changes to router support, they do not exhibit the complexity of elevated QoS approaches (policing, reservation signaling, admission control, etc.) and can be used to offer bounded delay to delay-sensitive applications.

Network QoS is often mistakenly regarded as a panacea for achieving high-quality networked applications. Much of this perception originates from the assumption that certain applications have very strict requirements and unless the network grants them with the required resources, they cannot run. However, there is evidence that applications are capable of operating within a range of resources that are available to them. An example is the audio and video transmission that make up the majority of the so-called real-time traffic on the Internet today. The old view of QoS was that an application would ask the network for what it needs and the network would set up a reservation and offer the requested guarantees. It appears that this approach to QoS is far too complex and does not scale; furthermore, the new reality of network communications makes end-to-end guarantees hard to achieve. As various technologies evolve, networking conditions become more and more heterogeneous: T1 and E1 lines, ATM circuits, LANs from 10BaseT up to Gigabit Ethernet, multicast, ever-growing wireless networks that by their nature have highly variable capacities, and a myriad of different hardware configurations. Given this situation, applications must be able to gracefully adapt to multiple environments. Applications can use several types of adaptation techniques to cope with changing network conditions:

1. Throughput adaptation. Multimedia flows are capable of adapting to the available bandwidth by varying their transmission rates. This is feasible due to the inherent ability of multimedia streams to operate with acceptable quality within a range of offered bit-rates by exploiting redundancies within the signal. Whatever the network circumstances (best-effort or QoS-enabled), each application should perform throughput or rate adaptation. Rate adaptation should be driven by appropriate congestion-control principles that strive to achieve fair use of available network resources. Congestion control should be promoted for two reasons [10]:

    - Congestion control can prevent the network from entering congestion collapse, a state in which the network is highly utilised by non-responsive flows, resulting in very little goodput for applications.
    - The network will soon develop mechanisms that penalise applications that do not perform congestion control, in the form of preferential dropping of their packets (e.g., RED) during periods of congestion.

2. Adaptation to delay and delay variation. Applications can adapt to variable delays, as long as these remain within reasonable ranges. There are certain levels of delay that can be tolerated by applications without conspicuous quality effects. When these levels are exceeded, applications can deploy mechanisms (e.g., playout buffers) that allow them to adapt to variable delays.

3. Adaptation to packet loss. Applications can employ several techniques to improve the resistance of their flows to packet loss. Such techniques can be either proactive (or direct, such as robust encoding and use of error protection), or reactive (or indirect, such as reduction of the transmission rate to ease congestion).

[[[The text between here and the start of 3.5.1 reads like notes for 3.5.1 – should it have been commented out? If it's not just notes, it needs to be filled out and put with the rest of the video-specific discussion.]]]

- Rate adaptation: How? (Should be TCP-friendly. Main trends: AIMD, general-AIMD, model-based a la TFRC)
- What's the influence of the video bit-rate dynamics:
  - Constant bit-rate (CBR) video. The video stream is encoded at a certain bit-rate. With CBR transmission mode, we get predictable traffic patterns, and makes network management much easier. However, any gain through statistical multiplexing is precluded. Furthermore, because of video content's inherent variability, CBR does not provide good video quality.
  - Unconstrained VBR. In this mode, the encoder operates regardless of any external constraint. This is achieved by selecting a constant quantisation scale throughout the encoding. It provides with an almost constant quality encoded video.
  - Constrained VBR. Several constraints might occur, such as, buffer size, encoding delay, etc. The encoder's effort is on maximising the quality given the constraints.
  - Feedback VBR. The encoder may possess knowledge about the state of the network and adjust to changes in the network, i.e., regulating its output bit-rate.
- What rate control techniques for video exist?
  - Source-based rate control. In source-based rate control, it is the responsibility of the sender application to adapt the video transmission rate. This can be accomplished by using implicit (congestion notifications signals, e.g. packet loss) or explicit (receiver's estimate of bandwidth) feedback.
  - Receiver-based rate control. The receiver is here responsible for selecting the appropriate rate for the stream. This may be achieved by the receiver adding or dropping layers of the audio or video video stream according to its estimate of the fair share of bandwidth in the network path.
  - Hybrid rate-control where both sender and receiver are involved.

  Methods of encoding and transmitting video:

  - Non-Scalable One-layer encoding. This method of coding is used in MPEG-1 and MPEG-2 Main Profile.
  - Scalable
  - Multiple encoded versions at different qualities
  - Multiple description coders
  - Hierarchical encoding - Layered Video. The audio or video stream is encoded as a number of cumulative layers of increasing quality. Each stream is transmitted separately.
  - Dynamic Rate shaping. In this method the rate of the encoded video stream is decreased by dropping higher level frequencies, colour components or frames.

- How can the application compensate for the effects of the different network-centric QoS parameters?
  - For delay and delay variation:
    o Adaptive playout techniques, see section3.5.1.2.
  - For packet loss:
    o pro-active: Use of more robust video coding. For example frame dependency during encoding may be avoided to
    avoid propagation of the effect of packet loss. o pro-active: Use redundancy (FEC). The main problem in this method is how to optimally select the bit-rate allocated to FEC coeds
    o Error concealment at the decoder
    o Retransmission
    o Some people even suggest TCP...

3.5.1 A review of common adaptation techniques for audio and video

This section presents a brief review of the most common adaptation techniques for audio and video streams.

3.5.1.1 Rate adaptation

Adapting the transmission rate of a media flow is a technique an application may use to react to changing availability of bandwidth. There are a number of different ways to achieve rate adaptation. Some of these can only be applied when media are pre-encoded and stored for later on-demand transmission; others can be used for both stored and live transmission.

The typical way to change the transmission rate for audio streams is to switch to another codec. This is due to the fact that typical audio codecs produce a Constant Bit Rate (CBR) stream. Layered codecs, especially for encoding music, have also been proposed.

There are numerous ways to adapt the output bit-rate of video:

- **Bandwidth negotiation.** The server-client system is able to make an estimation of the available capacity between the two ends prior to the transmission of video. After the negotiation phase, the server transmits a stream that closely matches the capacity characteristics of the path. This method is inflexible but simple, and is extensively used by several streaming products.

- **Multiple versions.** Most streaming products support multiple versions of the stream encoded at different bit-rates. The idea is that the server switches to the stream that best matches the client's capacity. Some streaming products support dynamic switching among the multiple streams. To achieve this, it is necessary to have synchronisation points in the bitstreams. The drawbacks of this method are that it complicates the encoding process and that it requires extra storage for the multiple encoded versions. Furthermore, the granularity of rate adaptation is limited by the number of available streams.

- **Control of the encoding parameters** is a common way of adapting to changing bandwidth availability. This method can be applied to both encoding of stored video (when, for example, a CBR stream may be required) and live video transmission where the encoding parameters are continuously updated by means of a *Rate Controller* at the encoder. A rate controller regulates the output bit-rate by appropriately selecting the value of the *quantisation parameter* Q by means of *rate-distortion* optimisation. Other parameters that can be used to increase or decrease the encoder's output rate include the frame skip rate, the resolution, and dropping of chrominance components.

- **Dynamic rate shaping** can be used to achieve reduction of a pre-encoded stream's bit-rate by efficiently eliminating several components of the encoded stream (e.g. high frequency coefficients, colour coefficients, and resolution change).

- **Scalable (hierarchical or layered) video encoding** refers to the encoding of a video stream into a set of cumulative sub-streams (layers) in which each layer is a refinement (i.e., improves the visual quality) of the previous layers. Most modern encoders (MPEG-2, MPEG-4, H.263+) support layered encoding. The flexible representation of layered flows, and the chances of achieving easy prioritisation of the video layers, are particularly appealing for network transport of video. Layered encoding can provide simple and efficient solutions to a number of problems:

  - Rate adaptation can be performed by adding or dropping layers according to the network conditions. Note that the granularity of adaptation is restrained by the granularity of the layers' bit-rates. This can be tackled by using sophisticated buffering [94, 96]. Alternatively, scalable two-layer codecs are quite popular: the base layer holds all the most important information, while the enhancement layer contains supplemental information and can use the remaining bandwidth.
  - Layered video is particularly useful for networks that employ some form of flow prioritisation and QoS. In this case higher priority can be given to the lower, more important layer(s).
  - Unequal error protection can be employed to protect the lower layers, as the impact of packet loss on quality is more severe for the lower layers than for the upper layers.

  There are many issues involved in layered encoding: (i) what partition method to use, (ii) the efficiency cost, (iii) how many layers to use, and/or (iv) how to distribute the bandwidth among the layers. Usually there is a performance penalty (in terms of PSNR) involved in scalable codecs in comparison to one-layer codecs, but the benefits can outweigh this. For example, in multicast congestion control, easy rate-adaptation and unequal error-protection can be more effective.

- **Multiple description coding (MDC)**. MDC features the encoding of a video stream into two independent streams. In contrast to layered encoding, where the presence of the lower layers is necessary for the higher layers to be decoded successfully, the streams produced by MDC (usually two) can be independently decoded. This property gives good protection against loss, since if either of the two streams is received then the video can be decoded. If both streams are received, then the quality of the decoded video is higher.

3.5.1.2 Adaptation to delay and delay variance

Variable end-to-end delays can be tackled by the use of a playout buffer. The type of application is the crucial factor here. Interactive applications can only afford a limited playout buffer depth. Other applications (live or on-demand streaming) can tolerate a playout buffer of several seconds. This technique can be adopted for both voice and video transmission. The idea of removing jitter is, instead of playing out media data as soon as they arrive, to delay their presentation by placing them into a receiver playout buffer for a certain amount of time, called the *playout delay*. The playout delay can be:

- *Fixed.* In this case, each data unit is delayed for a fixed time-interval T after it was generated. If a data unit (packet) arrives T *sec* after it was generated, it is discarded. Obviously, the value of T is a trade-off between resistance to jitter and interactivity.

- *Adaptive.* In this method, the play-out point of each data unit is adaptively changed according to the experienced delay. The changes of the play-out point are performed at media-suitable points — for

example, in the case of voice, at the beginning of a *talkspurt*[9]. If a media unit (speech sample, video frame) i is the first unit of a new talkspurt, then its playout time p_i, is calculated as:

$$p_i = t_i + \hat{d}_i + 4 * u_i \tag{3.1}$$

where t_i is the data unit's generation time, $\hat{d}_i$ is the end-to-end delay from the generation of the data to their playout, and u_i is the queuing delay the packet experienced in the network. In other words, $\hat{d}_i$ and u_i are estimates of the mean and variance of end-to-end delay. For all subsequent units j within the talkspurt, their respective playout points are calculated as:

$$p_j = p_i + t_i - t_j \tag{3.2}$$

Note that many other variations of adaptive playout algorithms have been proposed in the literature. [[[citation?]]]

3.5.1.3 Adaptation and resilience to packet loss

Packet loss is unavoidable in IP networks. Making use of techniques that increase the tolerance of media flows to loss and ameliorate the effects of packet loss on perceived quality is crucial in order for applications to achieve sustained quality. A significant number of approaches to adapting to different rates of packet loss and different loss patterns at the application layer have been suggested. These methods can complement each other at different parts of the application's end-to-end path, utilise different techniques, and fulfill different objectives. Adaptation to packet loss can be achieved either by using *proactive* or *indirect* solutions that aim at protecting the flow from potential packet loss, or by using *reactive* or *direct* techniques that do not protect the bitstream from packet loss per se but rather try to reduce the factors that cause packet loss or repair the part of the data that is damaged. Let us examine methods of achieving proactive and reactive adaptation to packet loss and analyse how these techniques are used on audio and video media flows. (Note that another categorisation of packet loss resilience techniques can be performed that is based on whether the technique is used at the sending or the receiving side of the application.)

Proactive packet loss resilience

- Use of robust encoding. In order to increase compression efficiency, encoding algorithms try to remove any redundancies in the signal by exploiting both spatial and temporal redundancies. Unfortunately, this makes the compressed bitstream more vulnerable to packet loss. For example, in video a packet loss can cause distortions that propagate through a series of successive frames until a new intra-coded frame arrives. A simple way to make the encoding process more robust is to allow the existence of more redundancy in the output signal. Such a method was used by McCanne in the *vic* video tool [76] by employing intra-coded frames only, together with conditional replenishment and block aging. Obviously this comes with the penalty of having less compression efficiency.

  Most recent video compression standards provide error resilient modes that are supported by a special bitstream syntax and some coding tools. H.263+ has several error resilience modes at the source coding level, including Slice-Structured Mode, Independent Segment Decoding Mode, and Reference Picture Selection (RPS) [139]. MPEG-4 provides several error resilience tools, such as re-synchronization markers, data partitioning, and reversible variable length codes (RVLC) [114]. A report of H.263+ and MPEG-4 error resilience performance can be found in [139, 115, 33]. All these tools, except the RPS mode in

[9] A talkspurt is defined as an interval with voice activity between two silent periods.

H.263+, are designed for intra-frame error control. They are most suitable for controlling errors that result from loss of small random data blocks within a video frame. They are not very effective in controlling errors that result from packet loss in video streaming.

- Forward error correction. Forward Error Correction (FEC) is a sender-based error-resilience technique that relies on the transmission of redundant packets from which the content of lost packets within the block can be recovered [91].

  - **Media-independent FEC** involves the application of block or algebraic codes on a codeword of k data packets to generate an additional $n-k$ check packets to aid the correction of losses. Several block coding techniques are proposed, the most popular being *parity* coding and *Reed-Solomon* coding. Parity coding is achieved by applying an exclusive-OR (XOR) operation across groups of packets to create parity packets. The Reed-Solomon encoding is based on the properties of polynomials, and has excellent correcting properties.

    The advantages of media-independent FEC are that it does not rely on the content of the packets, it is quite simple to implement, and it is not very computationally expensive. The drawbacks are increased decoder complexity and increased encoding delay (which can possibly affect interactivity). Furthermore, media-independent FEC increases the bandwidth requirement of the output bitstream.

  - The idea of **Media-specific FEC** is to piggyback the same data unit on one or more later packets. If a packet is lost, then another packet containing the lost data unit will be available for decoding. Let's consider the case of audio flows. Usually subsequent piggyback data units (secondaries) are encoded with a lower bandwidth, lower quality encoder than the first data unit (primary). The choice of the secondary encoding is particularly difficult as it is a trade-off between the bandwidth overhead and the encoding delay overhead. (Remember that low-delay encoders increase the redundancy bit-rate, while low-bit-rate encoders increase the computational complexity.)

    In the case of video, media-dependent FEC is often designed for a compressed bitstream and is performed before packetisation. H.263+ specifies this FEC mode for its bitstream [139].

    Unequal FEC can also be used to assign unequal amounts of FEC to the various segments of video data. This technique is called Priority Encoding Transmission (PET) [3]. PET requires that the priorities of video segments be specified.

    FEC is not always an ideal solution for video streaming. Recall that FEC increases the output bit-rate; as packet loss is highly correlated with congestion, FEC can be counterproductive. Of course, increasing the bit-rate can be avoided if the nominal bit-rate is partitioned between the compressed video data and FEC, but this comes at the expense of lower video quality. As packet loss conditions cannot be forecasted, it is difficult to know in advance what is the best (optimal) partition.

- Interleaving. Interleaving is a useful technique if the data units are smaller than the packet size, as in the case of speech transmission. Data units are re-sequenced prior to transmission such that originally adjacent units are separated by a certain distance, packetised, transmitted, and returned to their original sequence at the receiver. In this way a single packet loss results in multiple but small gaps in the reconstructed signal. This effect can be more easily patched by the human ear than can a single large gap. Large gaps usually affect the intelligibility of the speech and can affect whole phonemes. Interleaving does not increase bandwidth, but does increase latency.

Reactive packet loss resilience

- Congestion control. Congestion control is considered to be an indirect method of tackling packet loss. This is becoming more true as mechanisms that penalise unresponsive flows (e.g., RED) are being deployed

in routers. In a situation where packet loss is an indication of network congestion, if all flows react by dropping their transmission bit-rates, it is to be expected that packet loss will be reduced.

- Retransmission. Packets that are lost can be requested for retransmission by the sender application. Because retransmission involves a significant amount of delay, this method is obviously not appropriate for interactive applications. However, in the case of audio and video streaming, where there is a startup delay on the order of seconds, and higher values of jitter can be tolerated, retransmission becomes a viable approach. In fact, most of the commercial streaming applications on today's Internet rely heavily on retransmission of lost packets. Besides increasing playout delay, retransmission causes increased use of bandwidth. Note than FEC can be combined with retransmission.

- Error concealment. Error concealment is the process of reducing the effect of packet loss. This is a receiver-based technique, and can be used when sender-based error protection fails or is not present. Error concealment usually cannot provide the same level of resilience as FEC or retransmission, but can provide sufficient recovery at low loss rates and for small packet sizes.

  In the case of audio, error concealment techniques can be divided into three categories:

  - *Insertion-based* error concealment relies on the replacement of the missing packet with some other appropriate fill-in packet, such as [91]:
    - splicing together the two packets on either side of the lost packet. In general this method performs poorly.
    - Silence substitution, where the missing packet is replaced by a silence signal. This performs fairly well for short packet lengths and low bit-rates, but badly otherwise.
    - Noise substitution has been suggested in place of silence substitution, and seems to perform better.
    - Repetition involves replacing the data units lost with copies of the unit that was available just before the loss.
  - *Interpolation-based* techniques attempt to interpolate packets surrounding the lost packet to produce a replacement. Interpolation may be performed using: (i) waveform substitution, (ii) pitch waveform replication (an improvement over waveform substitution), and (iii) time-scale modification that allows the audio signal at each side of the loss to be stretched across the loss.
  - *Regeneration-based repair* tries to synthesise the lost audio signal by using knowledge of the compression algorithm to derive appropriate codec parameters.

Chapter 4

Measuring application quality: Tools and procedures

In this chapter we discuss recent work focused on the development of tools and methods for measuring application quality. Application quality is difficult to define in a straightforward and generic manner. It is a multi-parameter property linked with the nature of the application and the context of its use. In many cases, application quality is synonymous with the human user's degree of satisfaction; in others, it means the degree to which the application is capable of allowing its user to successfully complete a task.

Application quality has several different interpretations depending on a number of parameters described in chapter 2, and thus measures of quality need to be adaptive and flexible enough to reflect these parameters. In general, two different kinds of quality metrics are recognised. First, there are those that can be used off-line to gain an in-depth understanding of the specific application behaviour and requirements in terms of quality. Here quality might mean user perceptions or success in achieving the application's tasks and goals. These metrics can provide clues to how an application needs to be efficiently designed and engineered. In this category, several methods, techniques, disciplines, and combinations of these may be used:

- User studies and trials, including subjective quality assessment tests, questionnaires, specialised psychology tests, and interviews with user groups, application designers and service providers.
- Objective measures that use computational models to quantify the quality given a certain set of external application and environmental parameters (e.g., measurements of network performance).
- Results gained from similar experiences in the past.

A second kind of quality metrics are those that can be used to monitor the quality of an application during execution. Such metrics are equally important, as they provide invaluable feedback on the application's conditions when used in real-time. They can be also used to dynamically monitor the underlying transport mechanisms; to adapt, reschedule, reassign and reallocate resources; and to spot problematic modules on the application end-to-end path. Ideally, such metrics, given a snapshot of network conditions and the chosen QoS configuration (QoS service class, SLA agreement or other), should be able to evaluate the application performance in a concise and coherent way.

These two types of metrics complement one another. The background or off-line tools are believed to provide more accurate results because they may use multi-disciplinary techniques and exhaustive analysis of the results. The on-line quality tools are precious management and quality-monitoring mechanisms for real-time supervision

of a service. Since time constraints prohibit the use of computationally-expensive processing and analysis, they may sometimes fail to provide very accurate results. This provides a use for another important feature of the background tools: they can be used for alignment and calibration of the on-line models during their design and development process.

In the following sections we review models that belong to both these categories, analysing their main features and discussing the relative merits and weaknesses with respect to a number of applications and application components. We also present generic requirements and desirable features for the development of new measures.

4.1 Measuring the quality of video

As video becomes a fundamental part of advanced networking applications, being able to measure its quality becomes important at all stages, from the development of new video codecs to monitoring the quality of the transmission system. In order to appreciate and compare the performance of all involved components and devices, methods to assess and quantify video quality are proposed. In general, two classes of methods are available to measure video quality: *subjective tests*, in which human subjects are asked to assess or rank the viewed material, and *objective models*, which are computational models that measure the quality by comparing the original and distorted video sequences. Subjective tests may produce the most accurate ratings, but they require costly and complex setup and viewing conditions and are thus inflexible to use. Objective quality metrics, on the other hand, are based on mathematical methods — from quite simple models like PSNR, to sophisticated ones that exploit models of human visual perception and produce far more reliable results. While objective models of quality appear to be very promising, both methods are considered useful in the process of measuring the quality of video applications (and multimedia applications in general). Once successful standard objective models of video quality are developed, subjective tests can both complement them in evaluating application performance and be used to validate new objective models.

The next sections review work on subjective and objective video quality assessment, presenting their respective advantages and weak points.

4.1.1 Subjective video assessment

Subjective quality tests aim to capture the user's perception and understanding of quality. As we pointed out earlier, the user's perception of quality is not uni-dimensional, and it depends on many factors. As well as depending on the quality of the image, user perception is content-specific — it depends on whether or not the video material is interesting to the user. It has also been recognized that what determines quality also depends on the purpose of the interaction and the level of the user's engagement. The extent to which QoS is perceived as degraded depends on the real-world task that the user is performing. Furthermore, for many applications, the quality of the background sound is very important. Subjective quality ratings of the same video sequence are usually higher when it is accompanied by good-quality sound, as this may lower the viewer's ability to detect impairments [89]. For some applications, such as multimedia conferencing, users typically require higher audio quality relative to video quality. Perceived quality also depends on other factors like viewing distance, display size and resolution, and lighting conditions [2, 71, 75]. It is also worthwhile to mention the distinction between image quality as measured by mathematical procedures or computational models (i.e., the degree of distortion or difference between the original and reconstructed images) and the observed quality or *image fidelity*. It appears that images with higher contrast or slightly more colourful and saturated images appeal more to human viewers, even though, according to a strict mathematical interpretation of distortion (e.g., Signal-to-Noise Ratio (SNR)), they are distorted in comparison to the originals [31, 35].

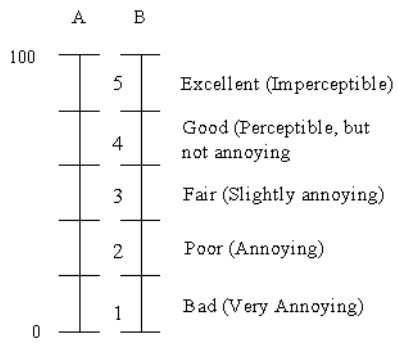

Figure 4.1: Video quality assessment scale used in subjective MOS tests

4.1.1.1 Procedures for subjective quality evaluation

Subjective quality assessment of video still remains the most reliable means of quantifying user perception. It is also the most efficient method of testing the performance of video codecs, human vision models and objective quality assessment metrics (section 4.1.2). This procedure, called Mean Opinion Scores (MOS), involves formal subjective tests where users are asked to rate the quality using a 5-point scale, as shown in Figure 1, with quality ratings ranging from *bad* to *excellent*.

ITU-R recommendation BT.500-10 [64] formalises this procedure by suggesting several experimental conditions, such as viewing distance, viewing conditions (room lighting, display features, etc.), selection of subjects and test material, and assessment and data analysis methods. Three testing procedures are most commonly used: *Double Stimulus Continuous Quality Scale*, *Double Stimulus Impairment Scale* and *Single Stimulus Continuous Quality Evaluation*.

Double Stimulus Continuous Quality Scale (DSCQS). In this method of subjective quality assessment, viewers are shown multiple sequence pairs, which consist of the original, "reference" sequence, and the reconstructed, "test" sequence. The sequences are relatively short in duration (8–10 seconds). The reference and test sequences are shown to the user twice in alternating fashion, with the order chosen randomly. Subjects do not know in advance which is the reference sequence and which is the test sequence. They rate the material on a scale ranging from "bad" to "excellent" (Figure 4.1), [[[Need close paren after "annoying" in "Good" level in this graphic]]] and the rating has an equivalent numerical scale from 0 to 100. The difference between the two ratings is taken for further analysis. This difference removes some rating uncertainties caused by the material content and viewers' experience. The DSCQS method is preferred when the quality of the reference and test sequences is similar; otherwise, subjects can easily spot the small differences in the quality of the two sequences. [[[This last sentence is puzzling — isn't the whole point for users to distinguish good from bad quality?]]]

Double Stimulus Impairment Scale (DSIS). In contrast to DSCQS, in this method the reference sequence is always presented before the test sequence, and there is no need for the pair to be shown twice. Rating of impairments is again on a 5-point scale, ranging from "very annoying" to "imperceptible" (Figure 4.1). This method is more useful for evaluating clearly visible impairments, such as noticeable artefacts caused by encoding or transmission.

Using short test sequences becomes a problem when we are interested in the evaluation of digital video systems operating over longer periods of time. These systems generate substantial quality variations that may

not be uniformly distributed over time. The DSCQS and DSIS methods were not designed to evaluate video transmission over packet networks like the Internet; because of the burstiness and non-deterministic behaviour of problems with packet-network video, the usefulness of these methods is limited. ¿From the user's point of view, perceived quality can vary significantly over time. If double stimulus (showing both the reference and test video sequences) is used for longer sequences, the time between comparable moments will be too long for them to be rated accurately. Furthermore, it is known that when the duration of the sequence is increased from 10 to 30 seconds, human memory is better for the more recent stimuli [4]. In other words, for longer sequences (over 10 seconds or so), the most recent parts of the sequence have a relatively greater contribution to the overall quality impression. This phenomenon, called *the recency effect*, has long been recorded in psychology literature [12], but it is difficult to quantify in subjective tests. Pearson [89] has discussed several higher-order effects that influence users' quality ratings when assessing video sequences of extended duration. What is needed here is a method able to dynamically capture users' opinions as the underlying network conditions or visual content complexity change.

Single Stimulus Continuous Quality Evaluation (SSCQE). In order to capture temporal variations in quality, viewers are shown a longer program, typically of 20–30 minutes duration. The reference is not presented, and viewers assess the instantaneously perceived quality by continuously adjusting a slider along the DSCQS scale (from "bad" to "excellent"). The slider can be implemented either as a hardware device [32] or as software. Instantaneous quality scores are obtained by periodically sampling the slider value, usually every 1–2 seconds. In this way, differences between alternative transmission configurations can be analysed in a more informative manner. The drawback of this method is that the accuracy of the user rating can be compromised by the cognitive load imposed by the task of moving the slider. As program content tends to have a significant impact on the SSCQE ratings, it becomes more difficult to compare scores from different test sequences. When a model is required to link instantaneously-perceived quality to an overall quality score calculated for the whole sequence, then the non-linear influence of good or bad parts within the sequence can be expressed by pooling methods like Minkowski power weighting [30]. Despite its attractiveness, the SSCQE method also exhibits several drawbacks, the most apparent of which is the impact of the recency effect on user judgements. Momentary changes in quality are quite difficult to track, making the stability and reliability of the derived results problematic.

4.1.2 Objective metrics of video quality

We stressed earlier that although subjective procedures for measuring video quality still constitute the most reliable method for gaining insight into the performance of digital video transmission systems, the complicated and costly setup of subjective tests makes this method unattractive for automating the assessment procedure. The involvement of human subjects in this process makes this approach unusable when the quality monitoring systems have to be embedded into practical processing systems. Quality metrics able to produce objectively obtained ratings present an attractive alternative.

Objective quality metrics have been the subject of research for several years. The first models were designed to work on analogue video transmission. However, the advent of digital manipulation and transmission of video means that video material is affected in a completely different way, leading to different types of impairments. This made it necessary to develop quality metrics that consider the impact of encoding and transmission in a digital system.

The simplest way to measure quality is to calculate the distortion at the pixel level. The *peak signal-to-noise ratio* (PSNR) measures the *mean squared error* (MSE) between the reference and test sequences, and has been extensively used by the image-processing community. Due to its simplicity it is still being used. While it is a

straightforward metric to calculate, it cannot describe distortions perceived by a complex and multi-dimensional system like the human visual system, so it fails to give good predictions in many cases. Recent research in image processing has focused on developing metrics that use models of the human visual system, while others exploit properties of the compression mechanism and assess the effect that known encoding artefacts have on perceived quality. In the following sections, we first briefly discuss the various types of distortions of compressed digital video transmission, then present a review of recent work on objective video quality metrics.

4.1.2.1 Impairments of digital video

End-to-end transmission of video material is subject to two sources of distortion. First the original content sequence is encoded (real-time or off-line) to reduce its (otherwise prohibitive) bandwidth requirement. As a result a first level of distortion caused by lossy encoding is introduced. The compressed bit-stream is then transmitted in packets over the network. There delay, delay-variation and packet loss make some information unavailable to the decoder, and further impairments occur. In the following paragraphs we describe the most common types of distortion introduced during the encoding and transmission of digital video.

Encoding Artefacts Most of the popular encoders rely on Motion Compensation (MC), block-based Discrete Cosine Transform (DCT) of blocks of pixels, and quantisation of the resulting transform coefficients. Quantisation is the main source of encoding distortions, although other encoding parameters, like frame dropping, also influence the perceived fidelity of the video. The main types of artefacts in a compressed video sequence [147] are the following:

- *Blocking effect or tiling.* Blockiness is defined in [8] as a distortion of the image characterised by the appearance of an underlying block encoding structure. Blockiness is caused by the independent quantisation of blocks, resulting in discontinuities at the boundaries of adjacent blocks. In other words, tiling creates false horizontal and vertical edges at the block boundaries. Due to its pattern, this is the most apparent visual distortion.
- *Blurring.* Blurring is a global distortion over the entire image, characterised by reduced sharpness of edges and spatial detail [8]. It is the result of the suppression of higher-frequency coefficients by a coarser quantisation.
- *Temporal edge noise or mosquito effect.* This is defined as a form of edge busyness characterised by time-varying sharpness (shimmering) at the edges of objects. This temporal artefact is the result of different coding of the same area of the image in subsequent frames.
- *Jagged motion.* Jagged motion is the result of poor motion estimation. When the residual error of motion prediction is large, then the quantisation error of the residual is also large.
- *Jerky motion.* Jerkiness is defined in [8] as originally smooth and continuous motion being perceived as a series of discontinuous images. This is due to lost motion energy when video is transmitted at lower frame rates (e.g., when frame dropping is adopted to keep within a bit-rate budget).
- Other artefacts are *colour bleeding* (smearing of colour between areas of strong chrominance difference), added *random noise*, and *chrominance mismatch* (due to the use of luminance motion vectors for chrominance motion compensation).

Some of the above effects are unique to block-based coding, while others are prevalent in other compression algorithms. In wavelet codecs, for example, there are no block-related artefacts, as the transform is applied to the entire image; however, blurring may become more noticeable.

Transmission Artefacts An important source of impairments is the transmission of the compressed video bitstream over the packet network. The bitstream is fragmented into a series of packets, which are then sent out to the destination. Two different types of impairments are attributed to the network behaviour: (i) packet loss, and (ii) end-to-end delay. When packets are lost they are unavailable to the decoder, while excessively delayed packets (high jitter) are worthless to the application. Therefore, both types have the same impact: data unavailability. The impact of such losses depends on the nature of the video encoder and the level of redundancy present in the compressed bitstream (for example, intra-coded bitstreams are more resilient to loss). For MC/DCT codecs, like MPEG, interdependencies of syntax information can cause an undesired effect in which the loss of a macroblock may corrupt subsequent macroblocks until the decoder can re-synchronise. This results in *error blocks* within the image; these bear no relationship to the rest of the image and usually contrast greatly with adjacent blocks. Obviously this has a major impact on perceived quality, usually a greater impact than coding artefacts. Another problem arises when blocks in subsequent frames are predicted from a corrupted macroblock — they will be damaged as well and this will cause a *temporal propagation of loss* until the next intra-coded macroblock is available.

4.1.2.2 Metrics based on human vision models

- ****Single channel models****
- ****Multiple channel models****

4.1.2.3 Metrics based on measuring features of perceptual distortions

Quality metrics that rely on models of the human visual system are potentially the most accurate. However, another category of quality metrics that use simpler vision models, but rely on knowledge about the impact of specific compression artefacts and transmission errors, is proved to produce almost equally reliable results. Due to the absence of complex vision models, these metrics allow for more computationally-efficient implementations. However, their drawback is that they work well only in particular application areas or with particular types of distortion. Objective quality models of this category feature the extraction of perceptually relevant, quality-related attributes of the signals (features); they measure the distance of the features in the test and reference images, and relate these distances to estimates of perceived quality. The following list represents the most influential work on this class of quality metrics (part of the discussion here is taken from [143]):

- Webster et al. [138] (1993) developed a video quality assessment system based on a combination of three low-level features. These features reflect the amount of lost and added energy in the spatial and temporal domains, and are calculated using spatial gradients and successive frames difference. These features are combined linearly to give a measure of video quality. These features were selected empirically from a number of candidates so as to yield the best correlation with subjective data for a given test set.

- The Institute for Telecommunication Sciences video quality metric [144] is based on the extraction and statistical manipulation of scalar spatio-temporal features from the original and degraded video sequences to obtain a single measure of distortion, and in essence is a progression from [138]. The extracted features closely represent the perceived effect of various distortions introduced from the encoder and/or the transmission system. These distortions may be present in the spatial as well as the temporal dimension. The most common spatial distortions appear in the form of blockiness (introduced by block-based DCT encoding), blurring or edge busyness (due to the absence of higher-frequency coefficients) and quantisation noise. Added motion energy may take the form of erroneous blocks that persist for a few frames, and lost motion energy (e.g., jerky video due to frame dropping). [[[Added motion energy takes the form of lost

motion energy?]]]The ITS model works on edge-enhanced (using Sobel filters) versions of the original and distorted sequences; gradient operators are used on the pixel domain to measure the magnitude and direction of the spatial gradient within spatio-temporal regions (S-T regions), usually $8 \times 8\, pixels \times 6\, frames$ in size. These gradients give an indication of the type and size of the distortion. For example, larger horizontal and vertical gradients in the distorted image are an indication of blocking or tiling artefacts. Here the lost edge energy in the output frame, measured by loss of diagonal spatial activity, is caused by the blurring impairment. The perceptual impairments within each S-T region are calculated using functions that model human visual masking. These impairments are then pooled using spatial (average of worst 5% of measured distortions) and then temporal collapsing (average over the clip's duration, which is 8–10 seconds) to obtain a single quality score for the video clip.

- Watson [135] extended a method previously proposed for optimal JPEG image compression, called DCTune[1], to the measurement of video quality. DCTune considers spatial sensitivity and masking, and in addition the video metric, called Digital Video Quality (DVQ), uses visibility thresholds for temporally varying quantisation noise and also measures temporal forward masking effects. The performance of this metric is discussed in [136].
- In [117], Tan et al. proposed an MPEG video quality metric. This metric computes the perceptual impairment based on contrast sensitivity and masking using spatial filtering and Sobel-operators. The masked error signal is then calculated and normalized. At a second stage, a cognitive emulator is used to simulate higher-level aspects of perception such as the delay and temporal smoothing effect of observer responses, the nonlinear saturation of perceived quality, and asymmetric behaviour with respect to quality changes from bad to good and vice versa. Although this tool requires a reference as input, the cognitive emulator was shown to improve the predictions of subjects SSCQE ratings.
- The quality assessment tool proposed by Hamada et al. [50] is based on a perceptual weighting of the coding noise using a three-layered design. Typical noise types from the compression are classified and weighted according to their characteristics. The local texture is analysed to compute the local degree of masking. Finally, a gaze-prediction stage is used to emphasize noise visibility in and around objects of interest. The measure of distortion is obtained from the PSNR computed on the weighted noise. This metric has been implemented in a system that permits real-time video quality assessment.

A complete review of work on objective video quality assessment can be found in [143] and the references therein.

4.1.3 Standardisation efforts

Efforts to design objective video quality models have generated an ongoing standardisation process that involves the specification of the essential features and requirements of objective quality assessment models, like the target video applications and the desired performance characteristics. Candidate models can then be assessed based on these specifications, with the aim of producing industry standards. We describe this standardisation work in the following.

4.1.3.1 Video Quality Experts Group

The Video Quality Experts Group (VQEG)[2] was formed in 1997 with the goal of evaluating and analysing the performance of video quality metrics in a variety of application areas (broadcast and cable television, multimedia,

[1] See http://vision.arc.nasa.gov/dctune/
[2] http://www.VQEG.org

etc.). The goal of this effort is to propose a suitable objective quality metric that would be standardised by relevant bodies, like the ITU. During the first phase of the project the emphasis was placed on evaluating distribution-class video metrics. Ten video quality assessment models (proponents), with the inclusion of PSNR for comparison purposes, were assessed and evaluated for their ability to provide predictions in agreement with subjective ratings.

1. Peak signal-to-noise ratio (PSNR, P0)
2. Centro de Pesquisa e Desenvolvimento (CPqD, Brazil, P1, August 1998)
3. Tektronix/Sarnoff (USA, P2, August 1998)
4. NHK/Mitsubishi Electric Corporation (Japan, P3, August 1998)
5. KDD (Japan, P4, model version 2.0 August 1998)
6. Ecole Polytechnique Federal Lausanne (EPFL, Switzerland, P5, August 1998)
7. TAPESTRIES (Europe, P6, August 1998)
8. National Aeronautics and Space Administration (NASA, USA, P7, August 1998)
9. Royal PTT Netherlands/Swisscom CT (KPN/Swisscom CT, The Netherlands, P8, August 1998)
10. National Telecommunications and Information Administration (NTIA, USA, P9, model version 1.0 August 1998)
11. Institut fur Nachrichtentechnik (IFN, Germany, P10, August 1998)

These proponents were tested over a wide range of video sequences and reference sequences. A large number of subjective tests were organised by independent labs, strictly adhering to the specifications of the ITU-R BT.500-8 [64] procedure for the DSCQS method of subjective evaluation (section 4.1.1.1). The analysis of the results showed that the performance of most of the proponents was statistically equivalent. Based on the analysis of results, VQEG concluded that currently it could not propose any model for inclusion in ITU Recommendations, and that further validation is required. However, the effort produced significant insights into the process of designing efficient objective video quality metrics and understanding the limitations of the current models. Details on the VQEG work and the evaluated proponents can be found in the 2000 final report [101]. Some of VQEG's future objectives have been presented in [100].

In the current phase of its work (phase II), VQEG has defined more precise areas of interest in order to obtain more accurate results than those obtained in phase I. These are:

- FR-TV: validation of full reference[3] objective methods for television
- RRNR-TV: validation of reduced reference and no reference methods for television
- RRNR-MM: validation of reduced reference and no reference methods for multimedia

[3]In the full reference (FR) method, both the reference and test sequences are required for the model to operate. In the reduced reference method (RR), a restricted amount of data from the reference is required to obtain the objective quality, while in the no reference method (NR), no reference data is required. The reduced reference and no reference methods decrease the transmission burden when on-line quality monitoring is required.

4.1.3.2 ITU Study Group 9

Study Group 9 (SG9) of ITU (see [61] for relevant information) has been working on preparing and maintaining recommendations for the delivery of audiovisual material (voice, sound, distribution television, video on demand, and other related services) over cable or hybrid networks. Among other efforts (for a full list of study problems see [62]), SG9 is working on standardisation efforts aimed at the quality evaluation of these services and networks (for example, the transport of audiovisual signals using IP), determination of quality parameters for television transport, and objective and subjective methods for the evaluation of audiovisual quality of conversational or distribution multimedia services.

4.1.4 Weaknesses of video quality assessment techniques

In the previous sections we described procedures and methods for measuring the quality of digital video. While these methods are very helpful for evaluating digital video components such as video codecs, monitoring transmission quality, and understanding psychophysical aspects of quality, there are still several limitations, especially if one focuses on Internet-based video services. Firstly, although subjective and objective methods can quite accurately measure the impairments of digital video, most of them work on short video sequences (typically, approximately 10 seconds duration). It is clear that a 10-second video sequence is not long enough to experience all the kinds of impairments that occur in a real Internet video application. This problem has been partially addressed by employing continuous assessment techniques ([51, 32]) and using temporal pooling methods such as Minkowski summation on objectively acquired quality scores [102]. Since objective video models that continuously assess quality have to be based on the corresponding subjective methods (e.g. SSCQE) for tuning, they inherit the same problems encountered with the subjective methods, as mentioned in section 4.1.1.1.

Secondly, the objective models described earlier cannot easily appreciate the impairments caused when digital video material is transmitted over an IP network. This happens because the original requirements were for the quality assessment of different transmission systems (like broadcast/cable TV transmission) and not the Internet. These transmission systems could provide an assured transmission channel, with bounded delay and different types of transmission errors, in contrast to what is experienced when transmitting video over the Internet. These models are designed to successfully track distortions of the visual content and not for other types of quality degradations, such those caused by large delays or extreme jitter. An objective metric of video quality needs to also develop methods that account for the joint effect of visual distortions (encoding distortions, packet loss) and non-visual distortions (delay and delay variation in the presentation of the video), and be able to measure the quality effects of all the Internet's idiosyncracies.

Thirdly, the quality judgements produced by these models are made solely for video. One may question this practice, since the video material (whether interactive or one-way) is rarely used on its own without audio. It is obvious that if judged jointly for the purpose of quality assessment, the inherent relations between the various media types involved in the application scenarios (especially between audio and video) may substantially alter the quality scores. Certain distortions may become more or less important, or new requirements arise (lip-synchronisation). The requirement for synchronisation of the audio and video components of a multimedia presentation is discussed in [52].

4.2 Measuring the quality of Internet audio

The quality of voice transmission over traditional networks like the PSTN can be measured quite accurately with the simple objective metrics specified in Recommendation G.712 [127] (one such metric is the over-used SNR). However, the evolution of voice applications over packet-switched networks involves different kinds of

technologies, such as low-bit-rate codecs, and different transmission characteristics. This renders traditional metrics inadequate for predicting a person's perception of speech clarity. Such metrics cannot account for a person's ability to adapt to missing time or frequency energy caused by encoding, and may indicate poor quality of the signal output even if its characteristics are not actually perceivable. For example, modern speech codecs use non-waveform encoding[4] and as a natural consequence the input signal is distorted considerably, resulting in very poor SNR. However, the nature of these codecs is such that the output characteristics of the voice would not cause significant discomfort to the listener, i.e. the encoded voice still sounds good. Thus the metrics need to calculate the signal clarity in the same context that users do, so that they can accurately represent and measure voice quality and clarity the way humans listeners perceive it. This section presents an overview of the most prominent methods of assessing speech quality. For a good introduction to voice quality metrics please refer to [7].

4.2.1 Mean Opinion Scores

As already mentioned in section 4.1.1.1, this method uses a large number of human listeners to produce subjective quality scores with statistical confidence (it considers the mean of the obtained quality opinions). The method is applied to both one-way and two-way (conversational) listening scenarios, and it is applied under a controlled testing environment (method of scoring, properties of voice samples used in tests, etc.). The overall MOS quality scores lie on a 1–5 scale[5], as shown in Figure 4.1. ITU Recommendation P.800 [58] describes the techniques for performing MOS, while ITU Recommendation P.830 [60] describes the methods for subjective evaluation of speech codecs. However, MOS quality assessment has several shortcomings:

- The setup of experiments can be quite costly and time-consuming, as it requires a large number of human subjects and a specialised and controlled environment.
- Because these tests directly involve humans and their subjective judgements, they are influenced by uncontrollable aspects of the subjects (mood, computer literacy, etc.), and thus cannot form a repeatable and consistent testing method.
- They are impractical if there is a requirement for frequent or on-line real-time testing. Such a requirement exists in the case of design and configuration of networks and appliances, and of routine quality monitoring of VoIP applications.

These drawbacks suggest that objective computational models that can automatically and repeatedly estimate the ongoing quality of speech are required to quantify the subjective clarity and quality of networked voice applications.

4.2.2 Objective methods of speech quality assessment

This section presents a brief description of several well-known models used to predict the quality of voice signals. We detail their merits and drawbacks and their suitability for specifying the network needs of IP voice. Note that these models can measure distortions caused by the process of encoding as well as distortions introduced by the transmission of speech over a transport network.

[4]Such codecs do not try to recreate the original waveform, but rather the perceptual properties of the speech signal (such as pitch) by attempting to model the human vocal tract.

[5]To recall, a quality rating of 5 corresponds to excellent quality, 4 to good, 3 to fair, 2 to poor and 1 to bad quality

4.2.2.1 Perceptual Speech Quality Measurement (PSQM)

The speech quality method called Perceptual Speech Quality Measurement (PSQM) [17] is a result of work at KPN Research in the Netherlands and has been approved by ITU-T Study Group 12 as Recommendation P.861 [59]. The method works on a pair of input and output (distorted) voice signals. The comparison is performed on time segments (frames) in the frequency domain. PSQM performs analysis of the spectral power densities of the input and output time frequencies, and applies comparisons based on aspects of the human hearing process such as sensitivity to loudness and frequency. The result is a score that measures the perceptual distance of the input and output signals; its values range from 0 (representing a perfect, undistorted signal) to 15-20 (high distortions). PSQM can accurately predict the speech clarity of voice signals that have been impacted by any of the following processes [59]:

- Waveform codecs (e.g., G.711, G.726)
- CELP-based codecs at bit-rates over 4 Kpbs (e.g., G.729a, G.723.1, G.728)
- Multiple bit rates of a codec
- Transcoding (the conversion from one digital format to another)

However, PSQM cannot give reliable results when the signal is impacted by factors due to transmission of speech over a network, like:

- Delay and delay variation
- Transmission errors (Bit Error Rate (BER), packet loss)
- Background noise
- Music as input signal
- CELP codes at bit-rates below 4 Kbps

The PSQM algorithm converts the actual physical signal to its mathematical representation based on the physiological properties of human perception. This is performed by three operations: (i) a *Fast Fourier Transform (FFT)* is performed on the input and output time domain signals, transforming them into the frequency domain. (ii) A *frequency warping and filtering* is done, where the frequency scale is warped to take into account human frequency sensitivities. Specific critical bands are used (PSQM defines 56 critical bands). (iii) An *intensity warping (compression)* is performed, where the intensity scale is warped to a loudness scale to represent human loudness sensitivities[6]. This process creates a mathematical representation of the acoustic signal based on the physiology of human hearing. In the next step, called *cognitive modelling*, the preprocessed input and output signals are compared to produce the PSQM score by evaluating the audible errors in the output signal. PSQM will disregard inaudible differences between input and output. As mentioned, the cognitive modelling will produce a PSQM score that ranges from 0 (perfect quality) up to 15 or higher (very bad).

4.2.2.2 Perceptual Speech Quality Measurement Plus (PSQM+)

While PSQM gained widespread acceptance, it was recognised that a major drawback was its inability to reliably report the impact of distortions generated from the network transport of speech, namely, those caused

[6] Depending on the loudness of the signal, voice distortions can be perceived differently by human listeners.

by packet loss and time clipping. In these cases, PSQM would output better quality than what human listeners would normally report. PSQM+ is a slight modification of ITU P.861 PSQM and improves its performance and reliability under conditions of packet loss and time clipping. The revised model was subsequently published by ITU [66]. To recall, PSQM measures the added or subtracted disturbance at each frame (time segment). As additive distortion (added energy) has larger impact than subtractive distortion (lost energy), the added disturbance is scaled up, thus producing a higher PSQM score (worse quality), while subtractive disturbance is scaled down, giving a lower score (higher quality). As packet loss and time clipping within a frame cause loss of energy, PSQM produced lower scores than it should have. PSQM+ fixes this by applying another scaling factor that scales up the disturbance. So, for added energy or low distortions due to the codec, PSQM and PSQM+ produce almost the same results. For large distortions due to lost energy (e.g., packet loss) PSQM+ will produce higher scores, correlating better with subjective results.

4.2.2.3 Measuring Normalised Blocks (MNB)

The MNB technique [132] was developed at the Institute for Telecommunications Sciences and represents an alternative to PSQM for measuring the perceptual distance between two voice signals. The algorithm works on perceptually modified input and output signals, and can be summarised as follows:

- During the *perceptual transformation*, the input and output signals are mapped to the frequency domain, and silence frames are removed.
- The input and output signals are input to the *Frequency Measuring Normalizing Block (FMNB)* where the time-frequency signals are integrated over time. The integrated measured difference between the input and output is subsequently subtracted from the output. The result is four FMNB measurements that result from the integration over four frequency bands.
- A similar process is performed in the *Time Measuring Normalizing Block (TMNB)*, where the time-frequency input and output are now integrated over frequency. The input is subtracted from the output, and the measured difference is subsequently subtracted from the output to produce a normalised output. The difference is then integrated over time to produce nine TMNBs.
- The independent FMNB and TMNB measurements are weight-combined to generate *Auditory Distance (AD)* values, which are then mapped to a range of MOS-correlated scores using a logistic function. [[[Is "logistic" the right word here?]]]

4.2.2.4 Perceptual Analysis Measurement System (PAMS)

The Perceptual Analysis Measurement System (PAMS) method [99, 24] was developed by British Telecommunications with similar objectives to PSQM+. It produces MOS-correlated scores in the 0-5 range. In brief, the algorithm performs the following steps:

- A perceptual modelling process (*auditory transform*) transforms the input and output signals into the time-frequency domain using Sekey filter banks [[[bands?]]] resulting in a time and frequency representation of perceived loudness (a *Sensation Surface*).
- An error sensation surface is produced by subtracting the input sensation surface from the output sensation surface to obtain audible differences. Positive errors correspond to noise or codec distortions while negative errors represent lost signal energy (due to loss, time-clipping, etc.).
- Audible errors are evaluated to predictive scores such that good correlation with subjective scores is achieved.

4.2.2.5 Perceptual Evaluation of Speech Quality (PESQ)

Perceptual Evaluation of Speech Quality (PESQ) [16, 98] is a collaborative effort by KPN Research and British Telecommunications; it leverages the best features of PSQM and PAMS. PESQ is intended to accurately measure the distortions of waveform and non-waveform codecs, transcoding, transmission errors, packet loss and time-clipping, etc. However, it has unknown accuracy for parameters and situations like delay, background noise, multiple talkers, codecs at bit-rates below 4 Kbps, and artificial speech or music as input. PESQ performs two kinds of processes:

- During *perceptual modelling*, the input and output signals are transformed into human perception representations using time-frequency mapping, frequency warping, and non-linear loudness scaling in a way that is compatible with human loudness sensitivity.
- A *cognitive modelling* process calculates two types of noise disturbance values; these are combined to produce a predicted MOS value. Cognitive modelling performs the following actions: calculation of time-frequency input-output difference, masking of small distortions, asymmetry processing (to differently weight negative and positive disturbances), detection of delay variation, and aggregation of disturbance values. The actual scores that are produced by PESQ are in the 0.5–4.5 range.

4.2.2.6 The E-model

The E-model [125, 124, 126], a recent ITU standard, is a computational model that predicts the subjective quality of networked speech based on transmission characteristics. It combines individual impairments resulting from both the properties of the signal and the characteristics of the transport medium into a single R rating that ranges from 0 to 100. This rating is then translated into the correlated MOS value. The relationship between R-ratings, speech quality and MOS values is shown in Figure 4.2. The R-rating is a linear combination of the perceived impairments converted to appropriate psycho-acoustic scales. The R value is given by the following formula:

$$R = (R_o - I_s) - I_d - I_e + A(1) \tag{4.1}$$

R_o represents the basic signal-to-noise ratio, and I_s accounts for loudness of connection and quantisation. Both these terms have to do with distortions of the voice signal not due to the network transmission itself. I_d and I_e are the factors that reflect the distortions caused by the network transmission; I_d captures the effect of delay while I_e captures the effect of information loss (that is, loss due to very low bit-rate encoding and packet loss). The value A represents an "advantage factor" that reflects the user's tolerance, or willingness to accept a certain amount of quality degradation. So, from the network point of view, the I_d and I_e factors are more interesting.

The I_d factor models the impact in conversational quality of end-to-end (mouth-to-ear, or "m2e") delay. It is further broken down into three terms: $I_d = I_{dte}(m2e, EL_2) + I_{dle}(m2e, EL_1) + I_{dd}(m2e)$, where $I_{dte}(m2e, EL_2)$ and $I_{dle}(m2e, EL_1)$ reflect the impact due to talker and listener echo, respectively. EL_1 and EL_2 represent the echo losses (in dB) at the points of reflection (these values depend on echo cancellation and are infinite if perfect echo cancellation is present). The term $I_{dd}(m2e)$ captures the degree of interactivity loss when end-to-end delay is large.

I_e is a measure of the distortion caused by the "loss of information or user data", which can be caused by low-bit-rate encoding or packet loss in the network or playback buffers. A good overview of the relation between the I_e R-factor and packet loss can be found in [74].

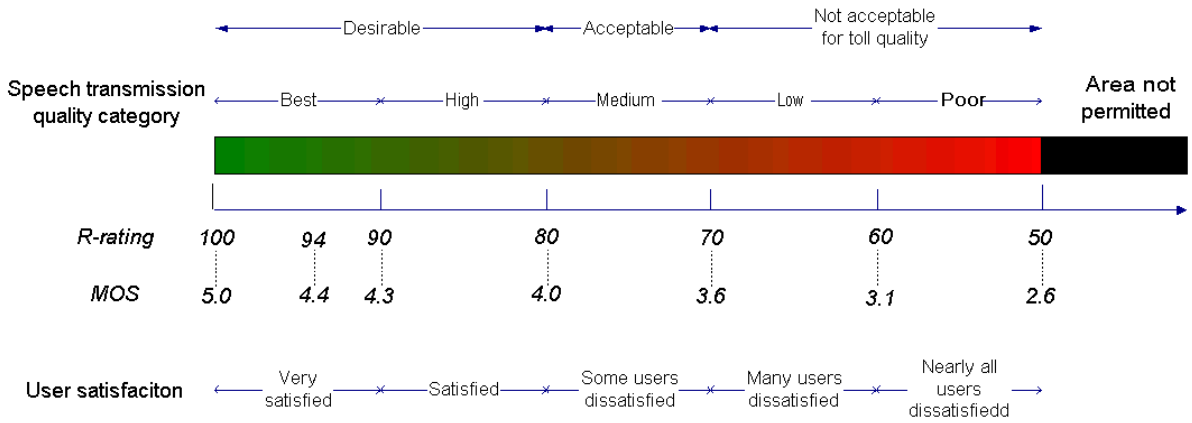

Figure 4.2: Mapping of R-rating values to MOS, speech transmission quality, and user satisfaction (Note that R values below 50 are not recommended for a speech transmission system).

Chapter 5

Conclusion

This report examined the qualitative and quantitative properties of quality of service requirements exhibited by advanced IP-based applications. We presented a taxonomy of applications based on several attributes: the task they have to perform, the degree of interactivity, and the kind of interactions (human-to-human, human-to-machine, or machine-to-machine). We discussed the characteristics of the application users, and the ways these factors directly or indirectly influence the perception of application quality. Another taxonomy was presented that examined applications' properties attributed to the nature and transport requirements of the participating media flows and their inherent capability to tolerate the impact of network transmission and adapt to changing conditions in the network.

We argued that the QoS needs of advanced applications can be studied from the viewpoint of the quality characteristics and requirements of their individual elementary data and media flows. We examined the quality requirements of these flows and the impact of network service on them, focusing on audio and video media flows and synchronisation of audio and video. Application adaptation was identified as crucial to application quality and to the well-being of the underlying network, and we reviewed common adaptation policies and decisions that audio-visual applications can use to cope with varying levels of QoS.

We presented a comprehensive review of procedures and tools that can be used to assess and measure the perceived quality of audio and video. There are two major approaches to performing quality assessment: subjective, by means of human Mean Opinion Scores (MOS), and objective, by using computational models that achieve high correlation with subjective scores. The advantages and disadvantages of these methods were described, and we suggested enhancements to them to account for the impact of Internet transmission.

Most of this part of the document dealt with the investigation of the quality requirements and idiosyncracies of audio-visual applications. For these applications, the degree of interactivity of the application is the principal determinant of requirements for end-to-end service. Users of significantly interactive applications, like VoIP or videoconferencing, can only tolerate low, bounded end-to-end delay and jitter. For non-interactive applications, like music or video streaming, such requirements are relaxed, and higher values of delay and delay variation can be comfortably accommodated. We also concluded that for these applications, loss of information, usually caused by packet loss in the network rising beyond a certain threshold, seems to cause more devastating quality deterioration than delay and jitter, as the effects of packet loss are directly perceived. Applications that do not have strict timing requirements may benefit from jitter compensation and error correction techniques designed to provide greater immunity to delay and packet loss. We described a plethora of proposed methods that an application can use, either in isolation or in a cooperative mode, to alleviate the effects of transmission over the network.

We argued that, for a number of reasons, the idea of a network that can always guarantee the performance requested by an application is far from becoming a reality in the foreseeable future. Most probably, the network will be able to provide some form of service discrimination for the different flows or some qualitative QoS guarantees. Within an ever-expanding heterogeneous network that can provide minimal guarantees, part of the burden of QoS must be placed on the application. The application is an ideal decision point where the most appropriate adaptation techniques can be selected and activated. We believe that under unfavourable network conditions, applications should have the right incentives to adapt. We examined what these incentives are. Non-adaptive application streams increase network congestion and are unfair to the rest of the flows on the network; this leads to worsening quality and may even cause network congestion collapse. In an ever-evolving heterogeneous Internet, applications that do not adapt are doomed to extinction. We presented a brief overview of the most common adaptation techniques that have been proposed in the literature. The aim of this document is to strongly encourage application designers and developers to build applications and services that make the most of what the network can offer instead of entirely relying on it.

This report is a working document and should not be considered complete. It is expected to be continuously updated with new and more focused analyses of the QoS needs of existing and emerging advanced applications. We hope that this document will contribute to a comprehensive understanding of what "application QoS" is and how it can be ensured by cooperative use of network- and application-level techniques.

Acknowledgements

A number of people made significant contributions to this document by providing feedback, comments and discussion. The members of the advisory committee (Amela Sadagic and Ben Teitelbaum (Advanced Network & Services), Jason Leigh (University of Illinois), and Magda El Zarki and Haining Liu (University of California, Irvine) provided continuous feedback and fruitful comments. Victor Reijs (HEAnet) provided very good comments and suggestions for improvement. Terry Gray and David Richardson (University of Washington) helped with HDTV issues. Stephen Wolf (Institute for Telecommunication Sciences) provided invaluable feedback on video quality. Hope Shu and Roch Guerin (University of Pennsylvania) offered comments on the structure and content of the document. Theodore Pagtzis (University College London) reviewed an early version of the document. Many thanks to Amela Sadagic and Ben Teitelbaum, who have been a continuous source of help, comments and support.

Bibliography

[1] The LIGO experiment. http://www.ligo.caltech.edu/.

[2] A. J. Ahumada and C. H. Null. Digital images and human vision. In A. B. Watson, editor, *Image Quality: a Multidimensional Problem*, pages 141–148. MIT Press, 1998.

[3] A. Albanese, J. Bloemer, J. Edmonds, M. Luby, and M. Sudan. Priority encoding transmission. *IEEE Transactions on Information Theory*, 42(6):1737–1747, November 1996.

[4] R. Aldridge, J. Davidoff, M. Ghanbari, D. Hands, and D. Pearson. Recency effect in the subjective assessment of digitally-coded television pictures. In *Fifth IEE International Conference on Image Processing and its Applications*, pages 336–339, Edingburgh, UK, July 1995.

[5] G. Almes, S. Kalidindi, and M. Zekauskas. A One-way Delay Metric for IPPM. RFC-2679, Internet Engineering Task Force (IETF), September 1999.

[6] G. Almes, S. Kalidindi, and M. Zekauskas. A One-way Packet Loss Metric for IPPM. RFC-2680, Internet Engineering Task Force (IETF), September 1999.

[7] J. Anderson. Methods for measuring perceptual speech quality. Agilent Technologies – White Paper. http://www.onenetworks.com/WhitePapers.asp.

[8] American National Standards Institute (ANSI). ANSI T1.801.02, Digital transport of video teleconferencing/video telephony signals – Performance terms, definitions and examples. Alliance for Telecommunications Industry Solutions, 1996.

[9] R. T. Apteker, J. A. Fisher, V. S. Kisimov, and H. Neishols. Video acceptability and frame rate. *IEEE Network*, 2(3):32–40, 1995.

[10] The TCP-Friendly Website at the Pittsburgh Supercomputing Center. http://www.psc.edu/networking/tcp_friendly.html.

[11] AT&T. Results of a subjective listening test for G.711 with frame erasure concealment. Contribution T1A1.7/99-016 to ITU-T, May 1999.

[12] A. D. Baddeley. *Working Memory*. Oxford University Press, Oxford, UK, 1986.

[13] H. Balakrishanan and S. Seshan. The congestion manager. RFC 3124, Jun 2001. draft-ietf-ecm-cm-04.txt.

[14] H. Balakrishnan, H. S. Rahul, and S. Seshan. An integrated congestion management architecture for internet hosts. In *ACM SIGCOMM 99*, Cambridge, MA., USA, Sep. 1999.

[15] W. Barfield and C. Hendrix. The effect of update rate on the sense of presence within virtual environments. *Virtual Reality: Research, Development, Applications*, 1(1):3–16, 1996.

[16] J. G. Beerends, A. W. Rix, A. P. Hekstra, and M. P. Hollier. Proposed Draft Recommendation P.862: Perceptual Evaluation of Speech Quality (PESQ), an Objective Method for End-to-end Speech Quality Assessment of Narrowband Telephone Networks and Speech Codecs. ITU-T, Study Group 12, May 2000.

[17] J. G. Beerends and J. A. Stemerdink. A perceptual speech quality measure based on a psychoacoustic sound representation. *Journal of Audio Engineering Society*, 42:115–123, March 1994.

[18] J.C. Bolot. End-to-end packet delay and loss behavior in the internet. In Deepinder P. Sidhu, editor, *SIGCOMM Symposium on Communications Architectures and Protocols*, pages 289–298, San Francisco, California, September 1993. ACM.

[19] J.C. Bolot and A. Vega Garcia. Control mechanisms for packet audio in the Internet. In *Proceedings of the Conference on Computer Communications (IEEE Infocom)*, San Fransisco, California, March 1996.

[20] M. S. Borella, D. Swider, S. Uludag, and G. B. Brewster. Internet packet loss: Measurement and implications for end-to-end QoS. In *Proceedings, International Conference on Parallel Processing*, August 1998.

[21] J. M. Boyce and R. D. Gaglianello. Packet loss effects on mpeg video sent over the public internet. In *ACM Multimedia 98*, Bristol, UK, September 1998.

[22] R. Braden, D. Clark, and S. Shenker. Integrated Services in the Internet Architecture: an Overview. RFC-1633, Internet Engineering Task Force (IETF), June 1994.

[23] R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin. Resource ReSerVation Protocol (RSVP) – Version 1 Functional Specification. RFC-2205, Internet Engineering Task Force (IETF), September 1997.

[24] PsyTechnics Group British Telecommunications. Pams usage guidelines, February 2000.

[25] B. Buxton. Integrating the periphery and context: A new taxonomy of telematics, 1995. http://www.dgp.toronto.edu/people/rroom/research/papers/bg_fg/bg_fg.html.

[26] C. Chafe, S. Wilson, R. Leistikow, D. Chisholm, and G. Scavone. A simplified approach to high quality music and sound over IP. In *Proceedings of the COST G-6 Conference on Digital Audio Effects (DAFX-00)*, Verona, Italy, December 2000.

[27] A. D. Clark. Modeling the effects of burst packet loss and recency on subjective voice quality. In *2nd IP-Telephony Workshop*, Columbia Univ., New York, USA, April 2001.

[28] D. D. Clark, S. Shenker, and L. Zhang. Supporting real-time applications in an integrated services packet network: Architecture and mechanism. In *ACM SIGCOMM*, pages 14–26, 1992.

[29] *DataGrid* project web page. http://eu-datagrid.web.cern.ch/eu-datagrid/.

[30] H. de Ridder. Minkowski-metrics as a combination rule for digital image coding impairments. In *SPIE Human Vision, Visual Processing and Digital Display*, volume 1666, pages 16–26, San Jose, CA, 1992.

[31] H. de Ridder, F. J. Blommaert, and E. A. Fedorovskaya. Naturalness anf image quality: Chroma and hue variation in color images of natural scenes. In *SPIE Human Vision, Visual Processing and Digital Display*, volume 2411, pages 51–61, San Jose, CA, 1995.

[32] H. de Ridder and R. Hamberg. Continuous assessment of image quality. *SMPTE Journal*, 106(2):123–128, February 1997.

[33] L. Ducla-Soares and F. Pereira. Error resilience and concealment performance for MPEG-4 frame-based video coding. *Signal Processing: Image Communication*, 14(6-8):447–472, May 1999.

[34] Ben Teitelbaum (Editor). Qbone architecture (v1.0). Internet2 QoS Working Group Draft, August 1999.

[35] E. A. Fedorovskaya, H. de Ridder, and F. J. Blommaert. Chroma variations and perceived quality of color images of natural scenes. *Color Research and Application*, 27(2):96–110, 1997.

[36] V. Firoiu, X. Zhang, and Y. Guo. Best effort differentiated services: Trade-off service differentiation for elastic applications. In *IEEE ICT 2001*, 2001.

[37] S. Floyd. Congestion control principles. Internet draft, Jan. 2000. draft-floyd-cong-01.txt.

[38] S. Floyd and K. Fall. Promoting the use of end-to-end congestion control in the internet. *IEEE/ACM Transactions on Networking*, 7(4):458–472, August 1999.

[39] S. Floyd and K. Fall. Promoting the use of end-to-end congestion control in the internet. *In IEEE/ACM Trans. on Networking*, 7(4):458–472, Aug. 1999.

[40] S. Floyd, M. Handley, J. Padhye, and J. Widmer. Equation-based congestion control for unicast applications. In *ACM SIGCOMM*, pages 43–56, Stockholm, Sweden, August 2000.

[41] P. Frossard. FEC performance in multimedia streaming. *IEEE Communications Letters*, 5(3):122–124, March 2001.

[42] P. Frossard and O. Verscheure. Joint source/FEC rate selection for quality-optimal MPEG-2 video delivery. *IEEE Transactions on Image Processing*, 10(2):1815–1825, December 2001.

[43] K. Fuduka, N. Wakamiya, and M. Murata. QoS mapping between user's preference and bandwidth control for video transport. In *Fifth Intl. Workshop on Quality of Service (IWQoS 97)*, IFIP, pages 291–302. Chapman & Hall, May 1997.

[44] G. Ghinea and J. P. Thomas. QoS impact on user perception and understanding of multimedia video clips. In *ACM Multimedia*, pages 49–54, Bristol, UK, September 1998.

[45] G. Ghinea, J. P. Thomas, and R. S. Fish. Multimedia, network protocols and users – bridging the gap. In *ACM Multimedia*, pages 473–476, Orlando, Florida, USA, October 1999.

[46] *GridPhyN* project web page. –. http://www.gridphyn.org.

[47] Nanoscale Science Research Group. The nanomanipulator. http://www.cs.unc.edu/Research/nano/cismm/nm/index.html.

[48] GTE Laboratories. Speech quality as a function of IP network performance. Contribution T1A1.5/98-022 to ITU-T, June 1998.

[49] GTE Laboratories. VoIP speech quality as a function of codec, manufacturer, jitter, and packet loss. Contribution T1A1.7/99-011 to ITU-T, May 1999.

[50] T. Hamada, S. Miyaji, and S. Matsumoto. Picture quality assessment system by three-layered bottom-up noise weighting considering human visual perception. *SMPTE Journal*, 108(1):20–26, 1999.

[51] R. Hamberg and H. de Ridder. Time-varying image quality: modelling the relation between instantaneous and overall quality. *SMPTE Journal*, pages 802–811, November 1999.

[52] D. Hands. Requirements for a multimedia perceptual model. ITU,Study Group 9 – Contribution 28, July 2001.

[53] T. Hanss. Internet2 Killer App or Dilbert's Nightmare? *EDUCAUSE review*, pages 17–25, May/June 2001.

[54] J. Heinanen, F. Baker, W. Weiss, and J. Wroclavski. Assured Forwarding PHB Group. RFC-2597, Internet Engineering Task Force (IETF), June 1999.

[55] C. Hendrix and W. Barfield. Presence within virtual environments as a function of visual display parameters. *Presence*, 5(3), 1996.

[56] P. Hurley, J.-Y. Le Boudec, P. Thiran, and M. Kara. ABE: Providing a low-delay service within best effort. *IEEE Network*, pages 60–69, May/June 2001.

[57] S. Iai, T. Kurita, and N. Kitawaki. Quality requirements for multimedia communication services and terminals – interaction of speech and video delays. In *Globecom '93 – IEEE Telecommunications Conference*, pages 394–398, Houston, TX, November 1993.

[58] Internation Telecommunication Union. Methods for subjective dedtermination of transmission quality. ITU-T Recommendation P.800, August 1996.

[59] Internation Telecommunication Union. Objective Quality Measurement of Telephone Band (300-3400 Hz) Speech Codecs. ITU-T Recommendation P.861, August 1996.

[60] Internation Telecommunication Union. Subjective performance assessment of telephone-band and wideband digital codecs. ITU-T Recommendation P.830, February 1996.

[61] International Telecommunication Union (ITU). ITU-T Study Group 9. http://www.itu.int/ITU-T/studygroups/com09/.

[62] International Telecommunication Union (ITU). Study Group 9 – List of questions under study (Study Period 2001 - 2004). http://www.itu.int/ITU-T/studygroups/com09/questions-sg9.doc.

[63] International Telecommunications Union (ITU). ITU-T-G.114, General Characteristics of International Telephone Connections and International Telephone Circuits , 1998.

[64] International Telecommunications Union (ITU). Recommendation BT.500.10, Methodology for the subjective assessment of the quality of television pictures, 2000.

[65] International Telecommunication Union (ITU-T). Network performance objectives for IP-based services. Series Y: Global Information Infrastructure. Draft New Recommendation Y.1541, October 2001.

[66] J. G. Beerends and E. J. Meijer and A. P. Hekstra. Improvement of the P.861 Perceptual Speech Quality Measure. Contribution COM 12-20 to ITU-T Study Group 12, December 1997.

[67] V. Jacobson. Multimedia conferencing on the Internet. ACM SIGCOMM '98 Tutorial lecture notes, August 1994.

[68] V. Jacobson and S. McCanne. Visual audio tool (vat) remote conferencing tool, July 1992. Manual page, available from ftp://ftp.ee.lbl.gov/conferencing/vat.

[69] V. Jacobson, K. Nichols, and K. Poduri. An Expedited Forwarding PHB Group. RFC-2598, Internet Engineering Task Force (IETF), June 1999.

[70] K. Jeffay, T. Hudson, and M. Parris. Beyond audio and video: Multimedia networking support for distributed, immersive virtual environments. In *Proceedings of the 27th EUROMICRO Conference*, pages 300–307, Warsow, Poland, september 2001.

[71] S. A. Klein. Digital images and human vision. In A. B. Watson, editor, *Image quality and image compression: A psychologist's viewpoint*, pages 73–88. MIT Press, 1993.

[72] V. Kumar, M. Korpi, and S. Sengodan. *IP Telephony with H.323*. Wiley, 2001.

[73] C. Lambrecht and O. Verscheure. Perceptual quality measure using a spatio-temporal model of the human visual system. In *Proceedings of SPIE Digital Video Compression: Algorithms and Technologies*, volume 2668, pages 450–461, San Jose, CA 1996.

[74] A. P. Markopoulou, F. A. Tobagi, and M. J. Karam. Assessment of VoIP Quality over Internet Backbones. In *IEEE Infocom*, 2002.

[75] J. B. Martens and V. Kayargadde. Image quality prediction in a multidimensional perceptual space. In *Intl. Conf. on Image Processing*, volume 1, pages 877–880, Lausanne, Switzerland, 1996.

[76] S. McCanne and V. Jacobson. vic : a flexible framework for packet video. In *ACM Multimedia*, pages 511–522, 1995.

[77] S. McCanne, V. Jacobson, and M. Vetterli. Receiver-driven layered multicast. In *ACM SIGCOMM*, pages 117–130, Palo Alto, CA, USA, August 1996.

[78] S. Mohamed, F. Cervantes-Perez, and H. Afifi. Integrating network measurements and speech quality subjective scores for control purposes. In *Proceedings of IEEE INFOCOM*, volume 2, pages 641–649, Anchorage, Alaska, April, 24-26 2001.

[79] K. Nakazono. Frame rate as a QoS paramater and its influenced in speech perception. *Multimedia Systems*, 6:359–366, 1998.

[80] UCL Network and Multimedia Research Group. MBone conferencing applications web page. http://www-mice.cs.ucl.ac.uk/multimedia/software/.

[81] K. Nichols, S. Blake, F. Baker, and D. Black. Definition of the Differentiated Services Field (DS Field) in the Ipv4 and Ipv6 Headers. RFC-2474, Internet Engineering Task Force (IETF), December 1998.

[82] K. Nichols, S. Blake, F. Baker, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss. An Architecture for Differentiated Services. RFC-2475, Internet Engineering Task Force (IETF), December 1998.

[83] K. Nichols, V. Jacobson, and L. Zhang. A two-bit differentiated services architecture for the internet, November 1997. ftp://ftp.ee.lbl.gov/papers/dsarch.pdf.

[84] Preservation & Encouragement of Barber Shop Quartet. http://www.spebsqsa.org/.

[85] L. F. Ortiz. Solving QoS in VoIP: A formula for explosive growth? BrookTrout Technology, 2001. http://www.brooktrout.com/news/in_the_news/0701_solve_qos.pdf.

[86] K. Park and R. Kenyon. Effects of network characteristics on human performance in a collaborative virtual environment. In *IEEE Virtual Reality*, pages 104–111, Houston, TX, March 1999.

[87] Nortel Networks Paul Coverdale. Multimedia QoS requirements from a user perspective. Presentation to ITU-T Workshop on QoS and user-perceived transmission quality in evolving networks, October 2001. http://www.itu.int/itudoc/itu-t/workshop/qos/s2p1.html.

[88] V. Paxson. End-to-end Internet packet dynamics. In *SIGCOMM Symposium on Communications Architectures and Protocols*, Cannes, France, September 1997.

[89] D. E. Pearson. Viewer response to time-varying video quality. In B. E. Rogowitz and T. N. Pappas, editors, *SPIE Human Vision and Electronic Imaging*, pages 16–25, Bellingham, WA, 1998.

[90] C. Perkins, L. Gharai, T. Lehman, and A. Mankin. Experiments with delivery of HDTV over IP networks. In 12th *International Packet Video Workshop (PV 2002)*, Pitsburgh, PA, USA, April 2002.

[91] C. Perkins, O. Hodson, and V. Hardman. A survey of packet loss recovery techniques for streaming audio. *IEEE Network*, 12(5):40–48, Sep.–Oct. 1998.

[92] The ViDe Large Scale Video Network Prototype. http://www.cavner.org/lsvnp/.

[93] V. Reijs. Perceived quantitative quality of applications, January 2002. http://www.heanet.ie/Heanet/projects/nat_infrastructure/perceived.html.

[94] R. Rejaie, M. Handley, and D. Estrin. Quality adaptation for congestion controlled video playback over the internet. In *Proceedings of ACM SIGCOMM*, pages 189–200, Harvard Univ., Cambridge, MA, USA, Aug. 31 - Sep. 3 1999.

[95] R. Rejaie, M. Handley, and D. Estrin. RAP: an end-to-end rate-based congestion control mechanism for real-time streams in the Internet. In *IEEE Infocom*, New York, NY, March 1999.

[96] R. Rejaie, M. Handley, and D. Estrin. Layered quality adaptation for internet video streaming. *IEEE Journal on Selected Areas of Communications (JSAC)*, 18(12):2530–2543, Dec. 2000. Special issue on Internet QOS.

[97] S. Rihs. The influence of audio on perceived picture quality and subjective audio-video delay tolerance. *MOSAIC Handbook*, pages 183–187, 1996.

[98] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra. PESQ - the new ITU Standard for End-to-end Speech Quality Assessment. In *AES 109th Convention*, September 2000.

[99] A. W. Rix and M. P. Hollier. The perceptual analysis measurement system for robust end-to-end speech quality assessment. *IEEE ICASSP*, June 2000.

[100] A. M. Rohaly and P. Corriveau et al. Video Quality Experts Group: Current results and future directions. VQEG Report, 2000. Available at ftp://ftp.crc.ca/crc/vqeg.

[101] A. M. Rohaly, J. Libert, P. Corriveau, and A. Webster (ed. committee). Final report from the Video Quality Experts Group on the validation of objective models of video quality assessment. Video Quality Experts Group (VQEG), April 2000. Available at ftp://ftp.crc.ca/crc/vqeg.

[102] A. M. Rohaly, J. Lu, N. R. Franzen, and M. K. Ravel. Comparison of temporal pooling methods for estimating the quality of complex video sequences. In B. Rogowitz T. N. Pappas, editor, *SPIE Human Vision and Electronic Imaging IV*, volume 3644, pages 218–225, 1999.

[103] E. Rosen, A. Viswanathan, and R. Callon. Multiprotocol Label Switching Architecture. RFC-3031, Internet Engineering Task Force (IETF), January 2001.

[104] P. Schopis. The reeality and mythology of Qos and H.323. Megaconference III, 2001. http://www.mega-net.net/megaconference/presentations/PaulSchopis.ppt.

[105] S. Shenker, C. Partridge, and R. Guerin. Specification of Guaranteed Quality of Service. RFC-2212, Internet Engineering Task Force (IETF), September 1997.

[106] V. Smotlacha. QoS oriented measurement in IP networks. Technical Report 17/2001, CESNET, December 2001.

[107] Society of Motion Picture and Television Engineers. Bit-serial digital interface for high-definition television systems. SMPTE-292M, 1998.

[108] M. Della Negra (Spokeperson). The CMS experiment. A Compact Muon Solenoid. http://cmsinfo.cern.ch.

[109] P. Jenni (Spokesperson). The ATLAS experiment. A Turoidal LHC ApparatuS. http://atlasinfo.cern.ch/Atlas/Welcome.html.

[110] R. Steinmetz. Human perception of jitter and media synchronization. *IEEE Journal on Selected Areas in Communications*, 14(1):61–72, January 1996.

[111] G. Strang and T. Nguyen. *Wavelets and Filter Banks*. Wellesley-Cambridge Press, Wellesley MA, first edition, 1996.

[112] L. Sun, G. Wade, B. Lines, and E. Ifeachor. Impact of packet loss location on perceived speech quality. In *2nd IP-Telephony Workshop*, pages 114–122, Columbia Univ., New York, USA, April 2001.

[113] A. S. Szalay. The Sloan Digital Sky Survey. *CiSE*, 1(2), 1999. http://www.sdss.org/.

[114] R. Talluri. Error resilient video coding in the MPEG-4 standard. *IEEE Communications Magazine*, 36:112–119, June 1998.

[115] R. Talluri, I. Moccagatta, Y. Nag, and G. Cheung. Error concealment by data partitioning. *Signal Processing: Image Communication*, 14(6-8):505–518, May 1999.

[116] K. T. Tan and M. Ghanbari. An objective video quality assessment system based on human perception. *IEEE Trans. on Cicruits and Systems for Video Technology*, 1913(7):15–26, February 1993.

[117] K. T. Tan, M. Ghanbari, and D. E. Pearson. An objective measurement tool for MPEG video quality. *Signal Processing*, 70(3):279–294, 1999.

[118] Telcordia Technologies. Impact of IP performance on Customer Perceived Quality. Contribution to ITU-T, Working Group T1A1.3, Project: T1A1-04, Document: T1A1.3/2000-041, October 2000.

[119] P. Teo and D. Heeger. Perceptual image distortion. In *Proceedings of ICIP-94 (IEEE International Conference on Image Processing)*, volume 2, pages 982–986, 1994.

[120] *AccessGrid* web page. http://www-fp.mcs.anl.gov/fl/accessgrid/.

[121] *Megaconference* web page. http://www.mega-net.net/megaconference/.

[122] The Washington University Internet HDTV web-page. http://www.washington.edu/hdtv/.

[123] UKERNA. Report of QoS Think Tank. ND/STG/QOS/DOC/010, version 1.0 release, The JNT Association, July 2001. http://www.ja.net/development/qos/qos_tt_report.pdf.

[124] International Telecommunication Union. Application of the Emodel: a planning guide. ITU-T G.108.

[125] International Telecommunication Union. The emodel, a computational model for use in transmission planning. ITU-T G.107.

[126] International Telecommunication Union. Transmission impairments due to speech processing. ITU-T Draft G.113 Appendix.

[127] International Telecommunication Union. Transmission performance characteristics of pulse code modulation. ITU-T Recommendation G.712.

[128] S. Vegesna. *IP Quality of Service*. Cisco Press, 2001.

[129] O. Verscheure, P. Frossard, and M. Hamdi. MPEG-2 video services over packet networks: Joint effect of encoding rate and data loss on user-oriented QoS. In *Proceedings of NOSSDAV 98*, pages 257–264, Cambridge, July 1998.

[130] O. Verscheure, P. Frossard, and M. Hamdi. User-oriented QoS analysis in MPEG-2 video delivery. *Journal of Real-Time Imaging (special issue on Real-Time Digital Video over Multimedia Networks)*, 5(5):305–314, October 1999.

[131] Video Development Initiative (ViDe). Video Conferencing Cookbook, ver. 2.0, April 2000. http://www.vide.gatech.edu/cookbook2.0/usecs.html.

[132] S. Voran. Objective estimation of perceived speech quality, Part I: Development of the measuring normalizing block technique; Part II: Evaluation of the measuring normalizing block technique. *IEEE Trans. on Speech and Audio Processing*, 7(4):383–390, July 1999.

[133] The Virtual Rooms Videoconferencing System (VRVS). http://www.vrvs.org/.

[134] Y. Wang, S. Wenger, J. Wen, and A. G. Katsaggelos. Error resilient video coding techniques. *IEEE Signal Processing Magazine (Special issue on Multimedia Communications over Networks)*, 17(4):61–82, July 2000.

[135] A. B. Watson. Toward a perceptual video quality metric. In *Proceedings of SPIE Human Vision and Electronic Imaging*, volume 3299, pages 139–147, Jan Jose, CA, 1998.

[136] A. B. Watson, J. Hu, J. F. McGowan, and J. B. Mulligan. Design and performance of a digital video quality metric. In *Proceedings of SPIE Human Vision and Electronic Imaging*, volume 3644, pages 168–174, Jan Jose, CA, 1999.

[137] The ViDeNet web page. http://www.cavner.org/videnet/.

[138] A. Webster. An objective video quality assessment system based on human perception. In *Human Vision, Visual Processing, and Digital Display IV*, pages 15–26, San Jose, CA, February 1993.

[139] S. Wenger, G. Knorr, J. Ott, and F. Kossentini. Error resilience support in H.263+. *IEEE Transactions on Circuits and Systems for Video Technology*, 8(7):867–877, November 1998.

[140] J. Widmer, R. Denda, and M. Mauve. A survey on TCP-friendly congestion control. *IEEE Network Magazine*, 2001. Special Issue on Control of Best Effort Traffic.

[141] S. Winkler. Issues in vision modeling for perceptual video quality assessment. *Signal Processing*, 78(2):231–252, 1999.

[142] S. Winkler. A perceptual distortion metric for digital color video. In *SPIE Human Vision and Electronic Imaging*, volume 3644, pages 175–184, San Jose, CA, USA, 1999.

[143] S. Winkler. *Vision models and quality metrics for image processing applications*. PhD thesis, Ecole Polytechnique Federal de Lausanne (EPFL), Lausanne, December 2000.

[144] S. Wolf and M. Pinson. Spatial-temporal distortion metrics for in-service quality monitoring of any digital video system. In *Proceedings of SPIE International Symposium on Voice, Video and Data Communications*, pages 175–184, Boston, USA, September 1999.

[145] J. Wroclawski. Specification of the ControlledLoad Network Element Service. RFC-2211, Internet Engineering Task Force (IETF), September 1997.

[146] M. Yajnik, S. Moon, J. Kurose, and D. Towsley. Measurement and modelling of the temporal dependence in packet loss. In *Proceedings of the Conference on Computer Communications (IEEE Infocom)*, New York, March 1999.

[147] M. Yuen and H. R. Wu. A survey of hybrid MC/DPCM/DCT video coding distortions. *Signal Processing*, 70(3):247–278, 1998.