

Voice over IP

15. Voice over IP: Speech Transmission over Packet Networks

J. Skoglund, E. Kozica, J. Linden, R. Hagen, W. B. Kleijn

The emergence of packet networks for both data and voice traffic has introduced new challenges for speech transmission designs that differ significantly from those encountered and handled in traditional circuit-switched telephone networks, such as the public switched telephone network (PSTN). In this chapter, we present the many aspects that affect speech quality in a voice over IP (VoIP) conversation. We also present design techniques for coding systems that aim to overcome the deficiencies of the packet channel. By properly utilizing speech codecs tailored for packet networks, VoIP can in fact produce a quality higher than that possible with PSTN.

15.1	Voice Communication	307
15.1.1	Limitations of PSTN.....	307
15.1.2	The Promise of VoIP.....	308
15.2	Properties of the Network	308
15.2.1	Network Protocols.....	308
15.2.2	Network Characteristics.....	309
15.2.3	Typical Network Characteristics.....	312
15.2.4	Quality-of-Service Techniques.....	313
15.3	Outline of a VoIP System	313
15.3.1	Echo Cancellation.....	314
15.3.2	Speech Codec.....	315
15.3.3	Jitter Buffer.....	315
15.3.4	Packet Loss Recovery.....	316
15.3.5	Joint Design of Jitter Buffer and Packet Loss Concealment.....	316
15.3.6	Auxiliary Speech Processing Components.....	316
15.3.7	Measuring the Quality of a VoIP System.....	317
15.4	Robust Encoding	317
15.4.1	Forward Error Correction.....	317
15.4.2	Multiple Description Coding.....	320
15.5	Packet Loss Concealment	326
15.5.1	Nonparametric Concealment.....	326
15.5.2	Parametric Concealment.....	327
15.6	Conclusion	327
	References	328

15.1 Voice Communication

Voice over internet protocol (IP), known as VoIP, represents a new voice communication paradigm that is rapidly establishing itself as an alternative to traditional telephony solutions. While VoIP generally leads to cost savings and facilitates improved services, its quality has not always been competitive. For over a century, voice communication systems have used virtually exclusively circuit-switched networks and this has led to a high level of maturity. The end-user has been accustomed to a telephone conversation that has *consistent* quality and low delay. Further, the user expects a signal that has a narrow-band character and, thus, accepts the limitations present in traditional solutions, limitations that VoIP systems lack.

A number of fundamental differences exist between traditional telephony systems and the emerging VoIP systems. These differences can severely affect voice quality if not handled properly. This chapter will dis-

cuss the major challenges specific to VoIP and show that, with proper design, the quality of a VoIP solution can be significantly better than that of the public switched telephone network (PSTN). We first provide a broad overview of the issues that affect end-to-end quality. We then present some general techniques for designing speech coders that are suited for the challenges imposed by VoIP. We emphasize multiple description coding, a powerful paradigm that has shown promising performance in practical systems, and also facilitates theoretical analysis.

15.1.1 Limitations of PSTN

Legacy telephony solutions are narrow-band. This property imposes severe limitations on the achievable quality. In fact, in traditional telephony applications, the speech bandwidth is restricted more than the inherent

limitations of narrow-band coding at an 8 kHz sampling rate. Typical telephony speech is band-limited to 300–3400 Hz. This bandwidth limitation explains why we are used to expect telephony speech to sound weak, unnatural, and lack crispness. The final connection to most households (the so-called local loop) is generally analog, by means of two-wire copper cables, while entirely digital connections are typically only found in enterprise environments. Due to poor connections or old wires, significant distortion may be generated in the analog part of the phone connection, a type of distortion that is entirely absent in VoIP implementations. Cordless phones also often generate significant analog distortion due to radio interference and other implementation issues.

15.1.2 The Promise of VoIP

It is clear that significant sources of quality degradation exist in the PSTN. VoIP can be used to avoid this distortion and, moreover, to remove the basic constraints imposed by the analog connection to the household.

As mentioned above, even without changing the sampling frequency, the bandwidth of the speech signal can be enhanced over telephony band speech. It is possible to extend the lower band down to about 50 Hz, which improves the base sound of the speech signal and has a major impact on the naturalness, presence, and comfort in a conversation. Extending the upper band to almost 4 kHz (a slight margin for sampling filter roll-off is necessary) improves the naturalness and *crispness* of the sound. All in all, a fuller, more-natural voice and higher intelligibility can be achieved just by extending the bandwidth within the limitations of narrow-band speech. This is the first step towards *face-to-face* communication quality offered by wide-band speech.

In addition to having an extended bandwidth, VoIP has fewer sources of analog distortion, resulting in the possibility to offer significantly better quality than PSTN

within the constraint of an 8 kHz sampling rate. Even though this improvement is often clearly noticeable, far better quality can be achieved by taking the step to wide-band coding.

One of the great advantages of VoIP is that there is no need to settle for narrow-band speech. In principle, compact disc (CD) quality is a reasonable alternative, allowing for the best possible quality. However, a high sampling frequency results in a somewhat higher transmission bandwidth and, more importantly, imposes tough requirements on hardware components. The bandwidth of speech is around 10 kHz [15.1], implying a sampling frequency of 20 kHz for good quality. However, 16 kHz has been chosen in the industry as the best trade-off between bit rate and speech quality for wide-band speech coding.

By extending the upper band to 8 kHz, significant improvements in intelligibility and quality can be achieved. Most notably, fricative sounds such as [s] and [f], which are hard to distinguish in telephony band situations, sound natural in wide-band speech.

Many hardware factors in the design of VoIP devices affect speech quality as well. Obvious examples are microphones, speakers, and analog-to-digital converters. These issues are also faced in regular telephony, and as such are well understood. However, since the limited signal bandwidth imposed by the traditional network is the main factor affecting quality, most regular phones do not offer high-quality audio. Hence, this is another area of potential improvement over the current PSTN experience.

There are other important reasons why VoIP is rapidly replacing PSTN. These include cost and flexibility. VoIP extends the usage scenarios for voice communications. The convergence of voice, data, and other media presents a field of new possibilities. An example is web collaboration, which combines application sharing, voice, and video conferencing. Each of the components, transported over the same IP network, enhances the experience of the others.

15.2 Properties of the Network

15.2.1 Network Protocols

Internet communication is based on the internet protocol (IP) which is a network layer (layer 3) protocol according to the seven-layer open systems interconnection (OSI) model [15.2]. The physical and data link

layers reside below the network layer. On top of the network layer protocol, a transport layer (OSI layer 4) protocol is deployed for the actual data transmission. Most internet applications are using the transmission control protocol (TCP) [15.3] as the transport protocol. TCP is very robust since it allows for retransmission

in the case that a packet has been lost or has not arrived within a specific time. However, there are obvious disadvantages of deploying this protocol for real-time, two-way communication. First and foremost, delays can become very long due to the retransmission process. Another major disadvantage of **TCP** is the increased traffic load due to transmission of acknowledgements and retransmitted packets. A better choice of transport layer protocol for real-time communication such as **VoIP** is the user datagram protocol (**UDP**) [15.4]. **UDP** does not implement any mechanism for retransmission of packets and is thus more efficient than **TCP** for real-time applications. On top of **UDP**, another Internet Engineering Task Force (**IETF**) protocol, the real-time transport protocol (**RTP**) [15.5], is typically deployed. This protocol includes all the necessary mechanisms to transport data generated by both standard codecs as well as proprietary codecs.

It should be mentioned that recently it has become common to transmit **VoIP** data over **TCP** to facilitate communication through firewalls that would normally not allow **VoIP** traffic. This is a good solution from a connectivity point of view, but introduces significant challenges for the **VoIP** software designer due to the disadvantages with deploying **TCP** for **VoIP**.

15.2.2 Network Characteristics

Three major factors associated with packet networks have a significant impact on perceived speech quality: delay, jitter, and packet loss. All three factors stem from the nature of a packet network, which provides no guarantee that a packet of speech data will arrive at the receiving end in time, or even that it will arrive at all. This contrasts with traditional telephony networks where data are rarely, or never, lost and the transmission delay is usually a fixed parameter that does not vary over time. These network effects are the most important factors distinguishing speech processing for **VoIP** from traditional solutions. If the **VoIP** device cannot address network degradation in a satisfactory manner, the quality can never be acceptable. Therefore, it is of utmost importance that the characteristics of the **IP** network are taken into account in the design and implementation of **VoIP** products as well as in the choice of components such as the speech codec. In the following sub-sections delay, jitter, and packet loss are discussed and methods to deal with these challenges are covered.

A fact often overlooked is that both sides of a call need to have robust solutions even if only one side is con-

nected to a poor network. A typical example is a wireless device that has been properly designed to be able to cope with the challenges in terms of jitter and packet loss typical of a wireless (**WiFi**) network which is connecting through an enterprise **PSTN** gateway. Often the gateway has been designed and configured to handle network characteristics typical of a well-behaved wired local-area network (**LAN**) and not a challenging wireless **LAN**. The result can be that the quality is good in the wireless device but poor on the **PSTN** side. Therefore, it is crucial that all devices in a **VoIP** solution are designed to be robust against network degradation.

Delay

Many factors affect the perceived quality in two-way communication. An important parameter is the transmission delay between the two end-points. If the latency is high, it can severely affect the quality and ease of conversation. The two main effects caused by high latency are annoying talker overlap and echo, which both can cause significant reduction of the perceived conversation quality.

In traditional telephony, long delays are experienced only for satellite calls, other long-distance calls, and calls to mobile phones. This is not true for **VoIP**. The effects of excessive delay have often been overlooked in **VoIP** design, resulting in significant conversational quality degradation even in short-distance calls. Wireless **VoIP**, typically over a wireless **LAN (WLAN)**, is becoming increasingly popular, but increases the challenges of delay management further.

The impact of latency on communication quality is not easily measured and varies significantly with the usage scenario. For example, long delays are not perceived as annoying in a cell-phone environment as for a regular wired phone because of the added value of mobility. The presence of echo also has a significant impact on our sensitivity to delay: the higher the latency, the lower the perceived quality. Hence, it is not possible to list a single number for how high latency is acceptable, but only some guidelines.

If the overall delay is more than about 40 ms, an echo is audible [15.6]. For lower delays, the echo is only perceived as an expected side-tone. For longer delays a well-designed echo canceler can remove the echo. For very long delays (greater than 200 ms), even if echo cancelation is used, it is hard to maintain a two-way conversation without talker overlap. This effect is often accentuated by shortcomings of the echo canceler design. If no echo is generated, a slightly higher delay is acceptable.

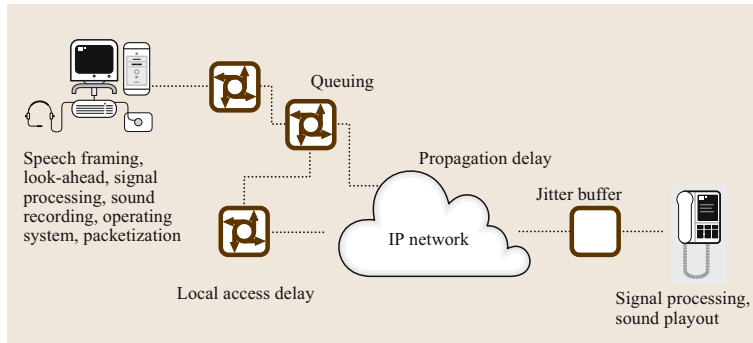


Fig. 15.1 Main delay sources in VoIP

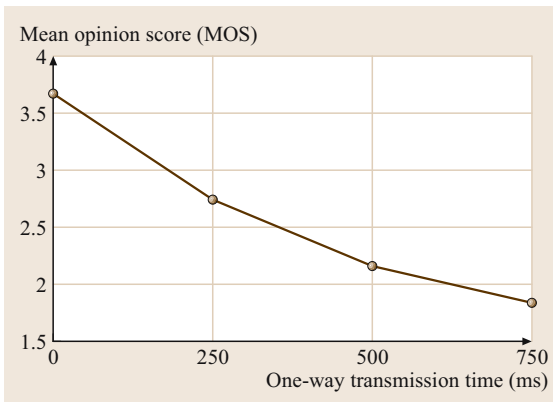


Fig. 15.2 Effect of delay on conversational quality from ITU-T G.114

The International Telecommunication Union – Telecommunication Standardization Sector (ITU-T) recommends in standard G.114 [15.7] that the one-way delay should be kept below 150 ms for acceptable conversation quality (Fig. 15.2 is from G.114 and shows the perceived effect on quality as a function of delay). Delays between 150 and 400 ms may be acceptable, but have an impact on the perceived quality of user applications. A latency larger than 400 ms is unacceptable.

Packet Loss

Packet losses often occur in the routers, either due to high router load or to high link load. In both cases, packets in the queues may be dropped. Packet loss also occurs when there is a breakdown in a transmission link. The result is data link layer error and the incomplete packet is dropped. Configuration errors and collisions may also result in packet loss. In non-real-time applications, packet loss is solved at the transfer layer by retransmission using TCP. For telephony, this is not a vi-

able solution since transmitted packets would arrive too late for use.

When a packet loss occurs some mechanism for filling in the missing speech must be incorporated. Such solutions are usually referred to as packet loss concealment (PLC) algorithms (Sect. 15.5). For best performance, these algorithms have to accurately predict the speech signal and make a smooth transition between the previous decoded speech and inserted segment.

Since packet losses occur mainly when the network is heavily loaded, it is not uncommon for packet losses to appear in bursts. A burst may consist of a series of consecutive lost packets or a period of high packet loss rate. When several consecutive packets are lost, even good PLC algorithms have problems producing acceptable speech quality.

To save transmission bandwidth, multiple speech frames are sometimes carried in a single packet, so a single lost packet may result in multiple lost frames. Even if the packet losses occur more spread out, the listening experience is then similar to that of having the packet losses occur in bursts.

Network Jitter

The latency in a voice communication system can be attributed to algorithmic, processing, and transmission delays. All three delay contributions are constant in a conventional telephone network. In VoIP, the algorithmic and processing delays are constant, but the transmission delay varies over time. The transit time of a packet through an IP network varies due to queuing effects. The transmission delay is interpreted as consisting of two parts, one being the constant or slowly varying network delay and the other being the rapid variations on top of the basic network delay, usually referred to as jitter.

The jitter present in packet networks complicates the decoding process in the receiver device because the

decoder needs to have packets of data available at the right time instants. If the data is not available, the decoder cannot produce continuous speech. A jitter buffer is normally used to make sure that packets are available when needed.

Clock Drift

Whether the communication end-points are gateways or other devices, low-frequency clock drift between the two can cause receiver buffer overflow or underflow. Simply speaking, this effect can be described as the two devices talking to each other having different time references. For example, the transmitter might send packets every 20 ms according to its perception of time, while the receiver's perception is that the packets arrive every 20.5 ms. In this case, for every 40th packet, the receiver has to perform a packet loss concealment to avoid buffer underflow. If the clock drift is not detected accurately, delay builds up during a call, so clock drift can have a significant impact on the speech quality. This is particularly difficult to mitigate in VoIP.

The traditional approach to address clock drift is to deploy a clock synchronization mechanism at the receiver to correct for clock drift by comparing the time stamps of the received RTP packets with the local clock. It is hard to obtain reliable clock drift estimates in VoIP because the estimates are based on averaging packet arrivals at a rate of typically 30–50 per second and because of the jitter in their arrival times. Consider for comparison the averaging on a per-sample basis at a rate of 8000 per second that is done in time-division multiplexing (TDM) networks [15.8]. In practice many algorithms designed to mitigate the clock drift effect fail to perform adequately.

Wireless Networks

Traditionally, packet networks consisted of wired Ethernet solutions that are relatively straightforward to manage. However, the rapid growth of wireless LAN (WLAN) solutions is quickly changing the network landscape. WLAN, in particular the IEEE 802.11 family of standards [15.9], offers mobility for computer access and also the flexibility of wireless IP phones, and are hence of great interest for VoIP systems. Jitter and effective packet loss rates are significantly higher in WLAN than in a wired network, as mentioned in Sect. 15.2.3. Furthermore, the network characteristics often change rapidly over time. In addition, as the user moves physically, coverage and interference from other radio sources—such as cordless phones, Bluetooth [15.10] devices, and microwave ovens—varies. The

result is that high-level voice quality is significantly harder to guarantee in a wireless network than a typical wired LAN.

WLANs are advertised as having very high throughput capacity (11 Mb/s for 802.11b and 54 Mb/s for 802.11a and 802.11g). However, field studies show that actual throughput is often only half of this, even when the client is close to the access point. It has been shown that these numbers are even worse for VoIP due to the high packet rate, with typical throughput values of 5–10% (Sect. 15.2.3).

When several users are connected to the same wireless access point, congestion is common. The result is jitter that can be significant, particularly if large data packets are sent over the same network. The efficiency of the system quickly deteriorates when the number of users increases.

When roaming in a wireless network, the mobile device has to switch between access points. In a traditional WLAN, it is common that such a hand-off introduces a 500 ms transmission gap, which has a clearly audible impact on the call quality. However, solutions are now available that cut that delay number to about 20 to 50 ms, if the user is not switching between two IP subnets. In the case of subnet roaming the handover is more complicated and no really good solutions exist currently. Therefore, it is common to plan the network in such a way that likelihood of subnet roaming is minimized.

Sensitivity to congestion is only one of the limitations of 802.11 networks. Degraded link quality, and consequently reduced available bandwidth, occurs due to a number of reasons. Some 802.11 systems operate in the unlicensed 2.4 GHz frequency range and share this spectrum with other wireless technologies, such as Bluetooth and cordless phones. This causes interference with potentially severe performance degradation since a lower connection speed than the maximum is chosen.

Poor link quality also leads to an increased number of retransmissions, which directly affects the delay and jitter. The link quality varies rapidly when moving around in a coverage area. This is a severe drawback, since a WLAN is introduced to add mobility and a wireless VoIP user can be expected to move around the coverage area. Hence, the introduction of VoIP into a WLAN environment puts higher requirements on network planning than for an all-data WLAN.

The result of the high delays that occur due to access-point congestion and bad link quality is that the packets often arrive too late to be useful. Therefore, the effective packet loss rate after the jitter buffer is typically significantly higher for WLANs than for wired LANs.

15.2.3 Typical Network Characteristics

Particularly VoIP communications that involve many hops are often negatively affected by delay, packet loss, and jitter. Little published work is available that describes network performance quantitatively. In one study towards improved understanding of network behavior, the transmission characteristics of the internet between several end-points were monitored over a four-week period in early 2002. Both packet loss and jitter were first measured as an average over 10 s periods. The maximum of these values over 5 min were then averaged over 7 d. Table 15.1 summarizes the results of these measurements. The table shows that significant jitter is present in long-distance IP communication, which affects speech quality significantly. Also, it was noted that, compared to Europe and the US, degradation in the quality of communications was more severe with calls to, from, and within Asia.

The ratio of infrastructure to traffic demand determines the level of resource contention. This, in turn, affects packet loss, delay, and jitter. With larger networks, packets can avoid bottlenecks and generally arrive within a reasonable time. When network conditions are less than ideal, communication quality deteriorates rapidly.

An informal test of the capacity of a wireless network was presented in [15.11]. The impact on packet loss and jitter of the number of simultaneous calls over a wireless access point with perfect coverage for all users is depicted in Figs. 15.3 and 15.4. Each call used the ITU-T G.711 codec [15.12] with a packet size of 20 ms which, including IP headers, results in a payload bandwidth of 80 kb/s. These results show that, for this access point, only five calls can be allowed if we do not allow packet loss. The results for this access point correspond to a bandwidth utilization factor of less than 10% percent. Interestingly, using a higher-compression speech codec did not increase the number of channels that can be handled. The reason is that access-point congestion depends much more on the number of packets the access point has to process than on the sum

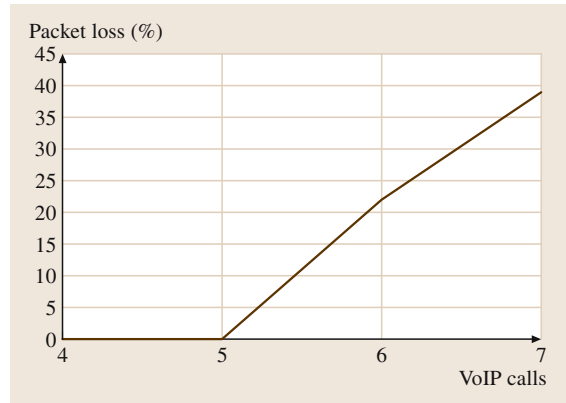


Fig. 15.3 Effect of access point congestion on the amount of packet loss as a function of the number of simultaneous VoIP calls through one access point (after [15.11])

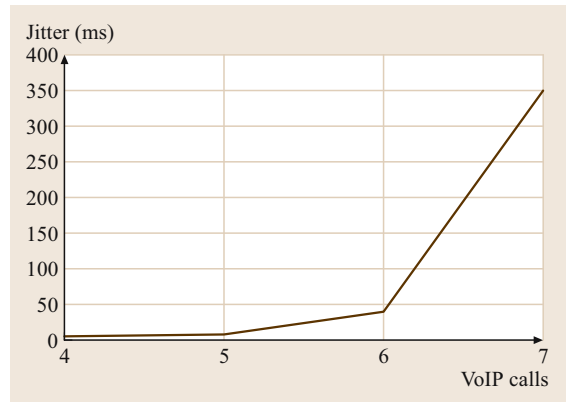


Fig. 15.4 Effect of access point congestion on network jitter as a function of the number of simultaneous VoIP calls through one access point (after [15.11])

of the user bit rates. Voice packets are small and sent very frequently, which explains the low throughput for voice packets. Because of this limitation it is common to put several voice frames into the same packet, which reduces the number of packets and hence increases the throughput. However, this results both in

Table 15.1 Results of international call performance monitoring. Long-term averages of short-term maximum values

Connection	Packet loss (%)	Roundtrip delay (ms)	Jitter (ms)	Hops
Hong Kong – Urungi, Xinjiang	50	800	350	20
Hong Kong – San Francisco	25	240	250	17
San Francisco – Stockholm	8	190	200	16–18

increased delay and in an increased impact of packet loss.

15.2.4 Quality-of-Service Techniques

Since network imperfections have a direct impact on the voice quality, it is important that the network is properly designed and managed. Measures taken to ensure network performance are usually referred to as quality-of-service (QoS) techniques. High quality of service can be achieved by adjusting capacity, by packet prioritization, and by network monitoring.

Capacity Management

Generally capacity issues are related to the connection to a wide-area network or other access links. It is uncommon that a local-area network has significant problems in this respect.

Prioritization Techniques

By prioritizing voice over less time-critical data in routers and switches, delay and jitter can be reduced significantly without a noticeable effect on the data traffic. Many different standards are available for implementing differentiated services, including IEEE 802.1p/D [15.13] and DiffServ [15.14]. In addition, the resource reservation protocol (RSVP) [15.15] can be used to reserve end-to-end bandwidth for the voice connection.

Network Monitoring

As the needs and requirements of networks continuously change, it is important to implement ongoing monitoring and management of the network.

Clearly, QoS management is easily ensured in an isolated network. Challenges typically arise in cases where the traffic is routed through an unmanaged network. A particularly interesting and challenging scenario is a telecommuter connecting to the enterprise network through a virtual private network. For more information on QoS, we refer the reader to the vast literature available on this topic, e.g. [15.16].

The QoS supplement developed in the Institute of Electrical and Electronics Engineers (IEEE) is called 802.11e [15.17] and is applicable to 802.11a, 802.11b and 802.11g. The development of this standard has been quite slow, and it likely will take time before significant deployment is seen. In the meanwhile, several vendors have developed proprietary enhancements of the standards that are already deployed. The introduction of some QoS mechanisms in WLAN environments will have a significant impact on the number of channels that can be supported and the amount of jitter present in the system. However, it is clear that the VoIP conditions in WLANs will remain more challenging than those of the typical wired LAN. Hence, particularly in the case of WLAN environments, QoS has to be combined with efficient jitter buffer implementations and careful latency management in the system design.

15.3 Outline of a VoIP System

VoIP is used to provide telephony functionality to the end user, which can communicate through various devices. For traditional telephony replacement, regular phones are used while so-called media gateways to the IP network convert the calls to and from IP traffic.

These gateways can be large trunking devices, situated in a telephony carrier's network, or smaller gateways, e.g., one-port gateways in an end user's home. An IP phone, on the other hand, is a device that looks very much like a regular phone, but it connects directly to an

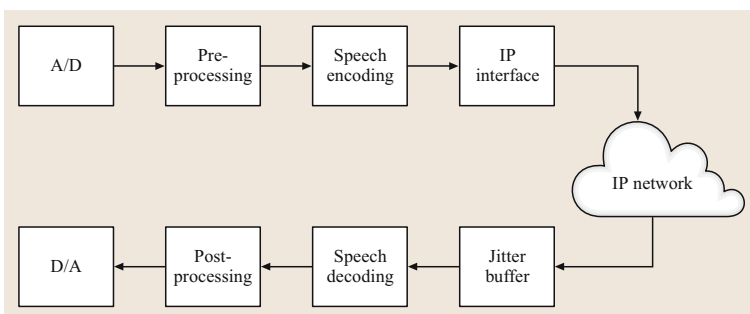


Fig. 15.5 A typical VoIP system

IP network. Lately, PCs have become popular devices for VoIP through services like Google Talk, Skype, Yahoo! Messenger and others. In this case, the phone is replaced by an application running on the PC that provides the telephony functionality. Such applications also exist for WiFi personal digital assistants (PDAs) and even for cell phones that have IP network connections. Hence, it is not an overstatement to say that the devices used in VoIP represent a changing environment and impose challenges for the speech processing components. Therefore, the systems deployed need to be able to cope with the IP network as described in the previous section, as well as the characteristics of the different applications.

A simplified block diagram of the speech processing blocks of a typical VoIP system is depicted in Fig. 15.5. At the transmitting end, the speech signal is first digitized in an analog-to-digital (A/D) converter. Next, a preprocessing block performs functions such as echo cancellation, noise suppression, automatic gain control, and voice activity detection, depending on the needs of the system and the end user's environment. Thereafter, the speech signal is encoded by a speech codec and the resulting bit stream is transmitted in IP packets. After the packets have passed the IP network, the receiving end converts the packet stream to a voice signal using the following basic processing blocks: a jitter buffer receiving the packets, speech decoding and postprocessing, e.g., packet loss concealment.

Next, we look closer at the major speech processing components in a VoIP system. These are echo cancellation, speech coding, jitter buffering and packet loss recovery. Further, we provide an overview of a number of auxiliary speech processing components.

15.3.1 Echo Cancellation

One of the most important aspects in terms of effect on the end-to-end quality is the amount of echo present during a conversation. An end user experiences echo by hearing a delayed version of what he or she said played back. The artifact is at best annoying and sometimes even renders the communication useless. An echo cancellation algorithm that estimates and removes the echo is needed. The requirements on an echo canceler to achieve good voice quality are very challenging. The result of a poor design can show up in several ways, the most common being:

1. audible residual echo, due to imperfect echo cancellation,

2. clipping of the speech, where parts of or entire words disappear due to too much cancellation,
3. poor double-talk performance.

The latter problem occurs when both parties attempt to talk at the same time and the echo canceler suppresses one or both of them leading to unnatural conversation. A common trade-off for most algorithms is the performance in double-talk versus single-talk, e.g., a method can be very good at suppressing echo, but has clipping and double-talk artifacts or vice versa [15.18].

Echo is a severe distraction if the round trip delay is longer than 30–40 ms [15.6]. Since the delays in IP telephony systems are significantly higher, the echo is clearly audible to the speaker. Canceling echo is, therefore, essential to maintaining high quality. Two types of echo can deteriorate speech quality: network echo and acoustic echo.

Network Echo Cancellation

Network echo is the dominant source of echo in telephone networks. It results from the impedance mismatch at the hybrid circuits of a PSTN exchange, at which the two-wire subscriber loop lines are connected to the four-wire lines of the long-haul trunk. The echo path is stationary, except when the call is transferred to another handset or when a third party is connected to the phone call. In these cases, there is an abrupt change in the echo path. Network echo in a VoIP system occurs through gateways, where there is echo between the incoming and outgoing signal paths of a channel generated by the PSTN network. Network echo cancelers for VoIP are implemented in the gateways.

The common solutions to echo cancellation and other impairments in packet-switched networks are basically adaptations of techniques used for the circuit-switched network. To achieve the best possible quality, a systematic approach is necessary to address the quality-of-sound issues that are specific to packet networks. Therefore, there may be significant differences between the “repackaged” circuit-switched echo cancelers and echo cancelers optimized for packet networks.

Acoustic Echo Cancellation

Acoustic echo occurs when there is a feedback path between a telephone's microphone and speaker (a problem primarily associated with wireless and hands-free devices) or between the microphone and the speakers of a personal computer (PC)-based system. In addition to the direct coupling path from microphone to speaker, acoustic echo can be caused by multiple reflections of

the speaker's sound waves back to the microphone from walls, floor, ceiling, windows, furniture, a car's dashboard, and other objects. Hence, the acoustic echo path is nonstationary. Acoustic echo has become a major issue in VoIP systems as more direct end-to-end VoIP calls are being made, e.g., by the very popular PC-based services.

There are few differences between designing acoustic echo cancelation (AEC) algorithms for VoIP and for traditional telephony applications. However, due to the higher delays typically experienced in VoIP, the requirements on the AEC are often more demanding. Also, wide-band speech adds some new challenges in terms of quality and complexity. Large-scale deployments in PC environments create demanding challenges in terms of robustness to different environments (e.g., various microphones and speakers) as well as varying echo paths created by non real-time operating systems, like Windows.

15.3.2 Speech Codec

The basic algorithmic building block in a VoIP system, that is always needed, is the speech codec. When initiating a voice call, a session protocol is used for the call setup, where both sides agree on which codec to use. The endpoints normally have a list of available codecs to choose from. The most common codec used in VoIP is the ITU-T G.711 [15.12] standard codec, which is mandatory for every end point to support for interoperability reasons, i.e., to guarantee that one common codec always exist so that a call can be established. For low-bandwidth applications, ITU-T G.729 [15.19] has been the dominant codec.

The quality of speech produced by the speech codec defines the upper limit for achievable end-to-end quality. This determines the sound quality for perfect network conditions, in which there are no packet losses, delays, jitter, echoes or other quality-degrading factors. The bit rate delivered by the speech encoder determines the bandwidth load on the network. The packet headers (IP, UDP, RTP) also add a significant portion to the bandwidth. For 20 ms packets, these headers increase the bit rate by 16 kbits/s, while for 10 ms packets the overhead bit rate doubles to 32 kbit/s. The packet header overhead versus payload trade-off has resulted in 20 ms packets being the most common choice.

A speech codec used in a VoIP environment must be able to handle lost packets. Robustness to lost packets determines the sound quality in situations where net-

work congestion is present and packet losses are likely. Traditional circuit switched codecs, e.g., G.729, are vulnerable to packet loss due to inter-frame dependencies caused by the encoding method. A new codec without interframe dependencies called the internet low-bit-rate codec (iLBC) has been standardized by the IETF for VoIP use [15.20]. Avoiding interframe dependencies is one step towards more robust speech coding. However, even codecs with low interframe dependencies need to handle inevitable packet losses and we will discuss that in more detail later.

Increasing the sampling frequency from 8 kHz, which is used for narrow-band products, to 16 kHz, which is used for wide-band speech coding, produces more natural, comfortable and intelligible speech. However, wide-band speech coding has thus far found limited use in applications such as video conferencing because speech coders mostly interact with the PSTN, which is inherently narrow-band. There is no such limitation when the call is initiated and terminated within the IP network. Therefore, because of the dramatic quality improvement attainable, the next generation of speech codecs for VoIP will be wide-band. This is seen in popular PC clients that have better telephony quality.

If the call has to traverse different types of networks, the speech sometimes needs to be transcoded. For example, if a user on an IP network that uses G.729 is calling a user on the PSTN, the speech packets need to be decoded and reencoded with G.711 at the media gateway. Transcoding should be avoided if possible, but is probably inevitable until there is a unified all-IP network.

It lies in the nature of a packet network that it, over short periods of time, has a variable-throughput bandwidth. Hence, speech coders that can handle variable bit rate are highly suitable for this type of channels. Variable bit rate can either be source-controlled, network-controlled, or both. If the encoder can get feedback from the network about current packet loss rates, it can adapt its rate to the available bandwidth. An example of an adaptive-rate codec is described in [15.21].

15.3.3 Jitter Buffer

A jitter buffer is required at the receiving end of a VoIP call. The buffer removes the jitter in the arrival times of the packets, so that there is data available for speech decoding when needed. The exception is if a packet is lost or delayed more than the length the jitter buffer is set to handle. The cost of a jitter buffer is an increase in the overall delay. The objective of a jitter buffer algorithm

is to keep the buffering delay as short as possible while minimizing the number of packets that arrive too late to be used. A large jitter buffer causes an increase in the delay and decreases the packet loss. A small jitter buffer decreases the delay but increases the resulting packet loss. The traditional approach is to store the incoming packets in a buffer (packet buffer) before sending them to the decoder. Because packets can arrive out of order, the jitter buffer is not a strict first-in first-out (FIFO) buffer, but it also reorders packets if necessary.

The most straightforward approach is to have a buffer of a fixed size that can handle a given fixed amount of jitter. This results in a constant buffer delay and requires no computations and provides minimum complexity. The drawback with this approach is that the length of the buffer has to be made sufficiently large that even the worst case can be accommodated or the quality will suffer.

To keep the delay as short as possible, it is important that the jitter buffer algorithm adapts rapidly to changing network conditions. Therefore, jitter buffers with dynamic size allocation, so-called adaptive jitter buffers, are now the most common [15.22]. The adaptation is achieved by inserting packets in the buffer when the delay needs to be increased and removing packets when the delay can be decreased. Packet insertion is usually done by repeating the previous packet. Unfortunately, this generally results in audible distortion. To avoid quality degradation, most adaptive jitter buffer algorithms are conservative when it comes to reducing the delay to lessen the chance of further delay increases. The traditional packet buffer approach is limited in its adaptation granularity by the packet size, since it can only change the buffer length by adding or discarding one or several packets. Another major limitation of traditional jitter buffers is that, to limit the audible distortion of removing packets, they typically only function during periods of silence. Hence, delay builds up during a talk spurt, and it can take several seconds before a reduction in the delay can be achieved.

15.3.4 Packet Loss Recovery

The available methods to recover from lost packets can be divided into two classes: sender-and-receiver-based and receiver-only-based techniques. In applications where delay is not a crucial factor automatic repeat request (ARQ) is a powerful and commonly used sender-and-receiver-based technique. However, the delay constraint restricts the use of ARQ in VoIP and other methods to recover the missing packets must be

considered. Robust encoding refers to methods where redundant side-information is added to the transmitted data packets. In Sect. 15.4, we describe robust encoding in more detail. Receiver-only-based methods, often called packet loss concealment (PLC), utilize only the information in previously received packets to replace the missing packets. This can be done by simply inserting zeros, repeating signals, or by some more-sophisticated methods utilizing features of the speech signal (e.g., pitch periods). Section 15.5 provides an overview of different error concealment methods.

15.3.5 Joint Design of Jitter Buffer and Packet Loss Concealment

It is possible to combine an advanced adaptive jitter buffer control with packet loss concealment into one unit [15.23]. The speech decoder is used as a *slave* in the sense that it decodes data and delivers speech segments back when the control logic asks for it. The new architecture makes the algorithm capable of adapting the buffer size on a millisecond basis. The approach allows it to quickly adapt to changing network conditions and to ensure high speech quality with minimal buffer latency. This can be achieved because the algorithm is working together with the decoder and not in the packet buffer. In addition to minimizing jitter buffer delay, the packet loss concealment part of the algorithm in [15.23] is based on a novel approach, and is capable of producing higher quality than any of the standard PLC methods. Experiments show that, with this type of approach, one-way delay savings of 30–80 ms are achievable in a typical VoIP environment [15.23]. Similar approaches have also been presented in, e.g., [15.24] and [15.25].

15.3.6 Auxiliary Speech Processing Components

In addition to the most visible and well-known voice processing components in a VoIP device there are many other important components that are included either to enhance the user experience or for other reasons, such as, reducing bandwidth requirements. The exploitation of such components depends on system requirements and usage scenarios, e.g., noise suppression for hands-free usage in noisy environment or voice activity detection to reduce bandwidth on bandlimited links.

Since no chain is stronger than its weakest link, it is imperative that even these components are designed in an appropriate fashion. Examples of such components include automatic gain control (AGC), voice activity de-

tection (**VAD**), comfort noise generation (**CNG**), noise suppression, and signal mixing for multiparty calling features. Typically, there is not much difference in the design or requirements between traditional telecommunications solutions and **VoIP** solutions for this type of components. However, for example **VAD** and **CNG** are typically deployed more frequently in **VoIP** systems. The main reason for the increased usage of **VAD** is that the net saving in bandwidth is very significant, due to the protocol overhead in **VoIP**. Also, an **IP** network is well suited to utilize the resulting variable bit rate to transport other data while no voice packets are sent.

VAD is used to determine silent segments, where packets do not need to be transmitted or, alternatively, only a sparse noise description is transmitted. The **CNG** unit produces a natural sound noise signal at the receiver, based either on a noise description that it receives or on an estimate of the noise. Due to misclassifications in the **VAD** algorithm clipping of the speech signal and noise bursts can occur. Also, since only comfort noise is played out during silence periods, the background signal may sound artificial. The most common problem with **CNG**, is that the signal level is set too low, which results in the feeling that the other person has dropped out of the conversation. These performance issues mandate that **VAD** should be used with caution to avoid unnecessary quality degradation.

Implementing multiparty calling also faces some challenges due to the characteristics of the **IP** networks. For example, the requirements for delay, clock drift, and echo cancelation performance are tougher due to the fact that several signals are mixed together and that there are several listeners present. A jitter buffer with low delay and the capability of efficiently handling clock drift offers a significant improvement in

15.4 Robust Encoding

The performance of a speech coder over a packet loss channel can efficiently be improved by adding redundancy at the encoding side and utilizing the redundancy at the decoder to fully or partly recover lost packets. The amount of redundancy must obviously be a function of the amount of packet loss. In this section we distinguish two classes of such so-called robust encoding approaches, *forward error correction* (**FEC**) and *multiple description coding* (**MDC**), and describe them in more detail. **MDC** is the more powerful of the two in the sense that it is more likely to provide graceful degradation.

such a scenario. Serious complexity issues arise since different codecs can be used for each of the parties in a call. Those codecs might use different sampling frequencies. Intelligent schemes to manage complexity are thus important. One way to reduce the complexity is by using **VAD** to determine what participants are active at each point in time and only mix those into the output signals.

15.3.7 Measuring the Quality of a VoIP System

Speech quality testing methods can be divided into two groups: subjective and objective. Since the ultimate goal of speech quality testing is to get a measure of the perceived quality by humans, subjective testing is the more relevant technique. The results of subjective tests are often presented as mean opinion scores (**MOS**) [15.26]. However, subjective tests are costly and quality ratings are relative from user to user. As a result, automatic or *objective* measuring techniques were developed to overcome these perceived shortcomings. The objective methods provide a lower cost and simple alternative, however they are highly sensitive to processing effects and other impairments [15.27, 28].

Perceptual evaluation of speech quality (**PESQ**), defined in ITU-T recommendation P.862 [15.29], is the most popular objective speech quality measurement tool. Even though **PESQ** is recognized as the most accurate objective method the likelihood that it will differ more than 0.5 **MOS** from subjective testing is 30% [15.30]. It is obvious that the objective methods do not offer the necessary level of accuracy in predicting the perceived speech quality and that subjective methods have to be used to achieve acceptable accuracy.

15.4.1 Forward Error Correction

In **FEC**, information in lost packets can be recovered by sending redundant data together with the information. Here we distinguish **FEC** methods from other methods that introduce redundancy in that the additional data does not contain information that can yield a better reconstruction at the decoder if no packets were lost. The typical characteristics of **FEC** are that the performance with some packet loss is the same as with no packet loss but that the performance rapidly deteriorates (a *cliff* effect) at losses higher than a certain critical packet loss

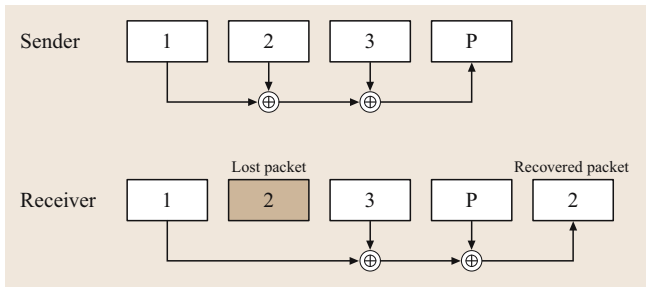


Fig. 15.6 FEC by parity coding. Every n -th transmitted packet is a parity packet constructed by bitwise XOR on the previous $n - 1$ packets

rate determined by the amount of redundancy. There are two classes of FEC schemes: *media-independent FEC* and *media-dependent FEC*.

Media-Independent FEC

In media-independent FEC, methods known from channel coding theory are used to add blocks of parity bits. These methods are also known as erasure codes. The simplest of these, called parity codes, utilize exclusive-or (XOR) operations between packets to generate parity packets. One simple example is illustrated in Fig. 15.6, where every n -th packet contains bitwise XOR on the $n - 1$ previous packets [15.31]. This method can exactly recover one lost packet if packet losses are at least separated by n packets. More-elaborate schemes can be achieved by different combinations of packets for the XOR operation. By increasing the amount of redundancy and delay it is possible to correct a small set of consecutively lost packets [15.32]. More-powerful error correction can be achieved using erasure codes such as

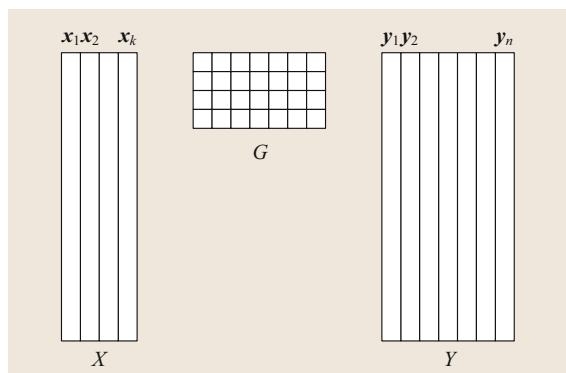


Fig. 15.7 FEC by an $RS(n, k)$ code-sending side. A sequence of k data packets is multiplied by a generator matrix to form n encoded packets

Reed-Solomon (RS) codes. These codes were originally presented for streams of individual symbols, but can be utilized for blocks (packets) of symbols [15.33]. The channel model is in this case an erasure channel where packets are either fully received or erased (lost). From k packets of data an $RS(n, k)$ code produces n packets of data such that the original k packets can be exactly recovered by receiving any subset of k (or more) packets. Assume each packet $\mathbf{x} = [x_1, x_2, \dots, x_B]^T$ contains B r -bit symbols $x_i = (b_i^{(1)}, b_i^{(2)}, \dots, b_i^{(r)})$, $b_i^{(j)} \in [0, 1]$.

The n packets can then be generated by (Fig. 15.7)

$$\mathbf{Y} = \mathbf{XG}, \tag{15.1}$$

where $\mathbf{X} = [x_1, x_2, \dots, x_k]$, $\mathbf{Y} = [y_1, y_2, \dots, y_n]$, and \mathbf{G} is a generator matrix. All operations are performed in the finite extension field $GF(2^r)$. The generator matrix is specific for the particular RS code. It is often convenient to arrange the generator matrix in systematic form, which yields an output where the first k packets are identical to the k uncoded packets and the other $n - k$ packets contain parity symbols exclusively. An example of constructing a generator matrix in systematic form is [15.33]

$$\mathbf{G} = \mathbf{V}_{k,k}^{-1} \mathbf{V}_{k,n}, \tag{15.2}$$

using the Vandermonde matrix

$$V_{i,j} = \alpha^{ij}, \tag{15.3}$$

where α is a generating element in $GF(2^r)$. The r -bit symbols are all elements in this field.

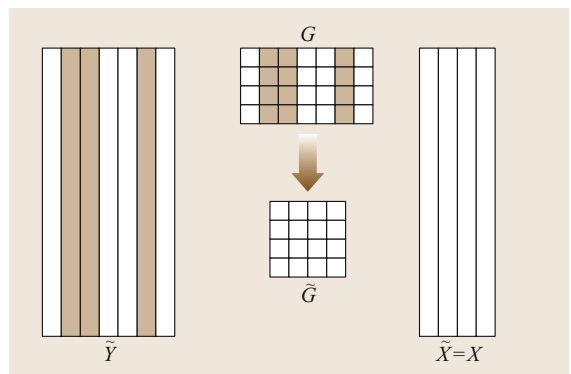


Fig. 15.8 FEC by an $RS(n, k)$ code-receiving side. Some packets are lost during transmission (indicated by the gray color). The decoder forms a matrix \tilde{G} from the columns corresponding to k correctly received packets. These packets are then multiplied with the inverse of \tilde{G} to recover \tilde{X} , the original sequence of k data packets

From k available packets the receiver forms \tilde{Y} , a subset of Y . The recovered packets are then calculated as

$$\tilde{X} = \tilde{Y}\tilde{G}^{-1}, \tag{15.4}$$

where \tilde{G} is formed by the k columns of G corresponding to the received packets. Figure 15.8 illustrates the decoding procedure. **RS** codes belong to a class of codes called maximum distance separable (MDS) codes that are generally powerful for many types of erasure channels but require a large n . For real-time interactive audio, delay is crucial and only short codes are feasible. Typical examples of **RS** codes utilized in **VoIP** are **RS**(5, 3) [15.34] and **RS**(3, 2) [15.35]. For the application of bursty erasure channels another type of codes, *maximally short codes* [15.36], have shown to require lower decoding delay than MDS codes.

A common way to decrease the packet overhead in **FEC** is to attach the redundant packets onto the information packet, a technique called *piggybacking*. Figure 15.9 depicts a case for an **RS**(5, 3) code, where the two parity packets are piggybacked onto the first two data packets of the next packet sequence.

Media-Dependent FEC

In media-dependent **FEC**, the source signal is encoded more than once. A simple scheme is just to encode the information twice and send the data in two separate packets. The extra packet can be piggybacked

onto the subsequent packet. To lower the overall bit rate, it is more common that the second description uses a lower rate-compression method, resulting in a lower quality in case a packet needs to be recovered. The latter method is sometimes referred to as low-bit-rate redundancy (LBR) and has been standardized by **IETF** [15.37], which actually also provides procedures for the first method. In the LBR context, the original (high-quality) coded description is referred to as the *primary encoding* and the lower-quality description is denoted the *redundant encoding*. Examples of primary and redundant encodings are G.711 (64 kbps)+**GSM** (13 kbps) [15.38] and G.729 (8 kbps)+**LPC10** (2.4 kbps) [15.35]. Figure 15.10 depicts the method for the case when the secondary encoding is piggybacked on the next primary encoding in the following packet. To safeguard against burst errors, the secondary description can be attached to the m -th next packet instead of the immediate subsequent packet, at the cost of a decoding delay of m packets. Even better protection is obtained with more redundancy, e.g., packet n contains the primary encoding of block data block n and redundant encodings of blocks $n - 1, n - 2, \dots, n - m$ [15.38]. Although media-dependent **FEC** seems more popular, erasure codes are reported to have better subjective performance at comparable bit rates [15.35].

It is important to point out that most practical systems that today use **FEC** simply put the same encoded frame in two packets, as mentioned above. The rea-

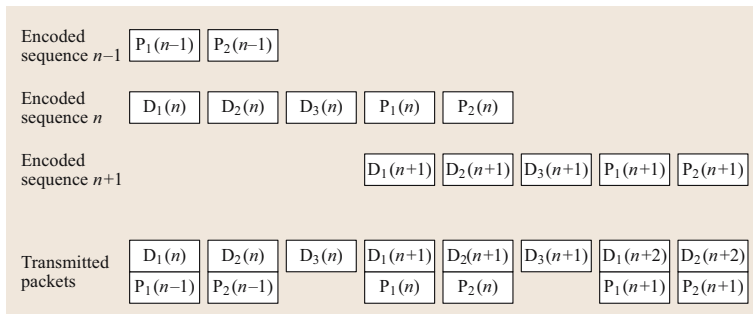


Fig. 15.9 Piggybacking in an **RS**(5, 3) **FEC** system. The parity packets of sequence n are attached to the data packets of sequence $n + 1$.

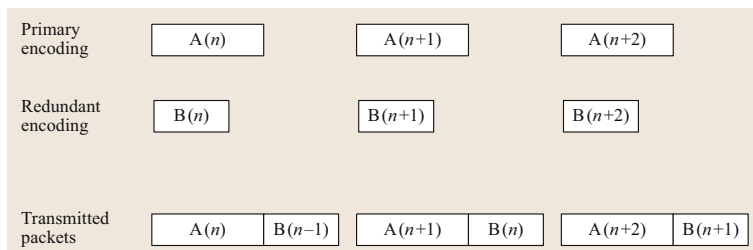


Fig. 15.10 Media-dependent **FEC** proposed by **IETF**. The data is encoded twice, and the additional redundant encoding is piggybacked onto the primary encoding in the next packet

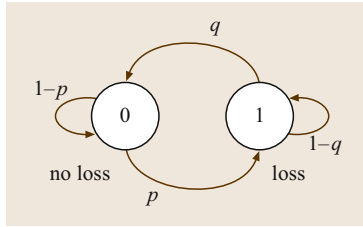


Fig. 15.11
The Gilbert two-state model of a bursty packet loss channel

son for doing so is due to the fact that a lower bit rate secondary coder is typically more complex than the primary encoder and the increase in bit rate is preferred to an increase in complexity.

Adaptive FEC

The amount of redundancy needed is a function of the degree of packet loss. Some attempts have been made to model the loss process in order to assess the benefits of adding redundancy. In [15.38] Bolot et al. used LBR according to the IETF scheme mentioned in the last paragraph and a Gilbert channel model (a Markov chain model for the dependencies between consecutive packet losses, Fig. 15.11) to obtain expressions for the perceived quality at the decoder as a function of packet loss and redundancy. In the derivations the perceived quality is assumed to be an increasing function of the bit rate. They propose a design algorithm to allocate redundancy that functions even in the case that there are only a handful of low bit rate secondary encodings to choose from. In the case where the channel packet loss can be estimated somewhere in the network and signalled back to the sender, e.g., through RTCP (an out-of-band signaling for RTP [15.5]), their algorithm can be utilized for adaptive FEC. Due to the adaptation it is possible to obtain a more graceful degradation of performance than the typical collapsing behavior of FEC.

15.4.2 Multiple Description Coding

Multiple description coding (MDC) is another method to introduce redundancy into the transmitted description to counter packet loss. Compared to erasure codes, the method has the advantage that it naturally leads to graceful degradation. It optimizes an average distortion criterion assuming a particular packet loss scenario, which can in principle consist of an ensemble of scenarios. Disadvantages of the multiple description coding technique are that it is difficult to combine multiple description coding with legacy coding systems and that changing the robustness level implies changing the entire source coder.

MDC Problem Formulation

Consider a sampled speech signal that is transmitted with packets. To facilitate the derivations that follow below, the speech signal contained in a packet is approximated as a weakly stationary and ergodic process that is independent and identically distributed (i.i.d.). We denote this speech process by X (thus, X denotes a sequence of random variables that are i.i.d.). We note that a process X with the aforementioned properties can be obtained from speech by performing an appropriate invertible transform. Optimal encoding of X over a channel facilitating a given rate R requires the *one* description of the source, which, when decoded at the receiver end, achieves the lowest achievable distortion $E[d(X, \hat{X})]$, where \hat{X} is the reconstructed process. Varying the rate leads to a lower bound on all achievable distortions as a function of the rate, i. e., the distortion-rate function $D(R)$.

An extension of the problem formulation, to the case where several channels are available, is not entirely intuitive. We consider here only the case of two channels, $s \in \{1, 2\}$, each with rate R_s and corresponding description I_s . The channels are such that they either do or do not work. (This corresponds to the cases where the packet either arrives or does not arrive.) Thus, we can distinguish four receiver scenarios: we receive both descriptions, we receive description I_1 solely, we receive description I_2 solely, or we receive no description. A particular formulation of the objective of multiple description coding is to find the two descriptions that minimize the central distortion $E[d(X, \hat{X}_0)]$, with constraints on the side distortions, $E[d(X, \hat{X}_s)] < D_s$, where \hat{X}_0 and \hat{X}_s are *central* and *side* reconstruction values obtained from both descriptions and a single description, respectively.

Figure 15.12 illustrates the operation of multiple description coders. The encoders f_s map the speech signal X to two descriptions I_s , each of rate R_s . From these two descriptions, three reconstruction values from three decoders can be obtained, depending on which combi-

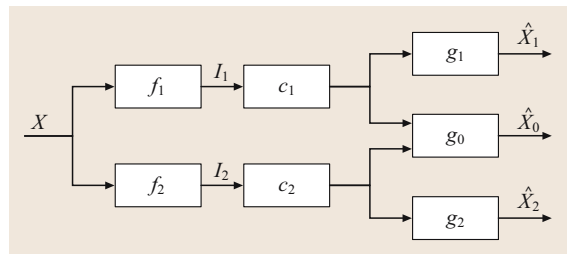


Fig. 15.12 Block diagram of a multiple description system with two channels c_s , two encoders f_s , and three decoders g_0 and g_s . Each channel is denoted by an index $s \in \{1, 2\}$

nation of descriptions is used. When both descriptions are received, the central decoder is active and the reconstructed value is $\hat{X}_0 = g_0(I_1, I_2)$. When only one description is received, the corresponding side decoder is active and the reconstructed value is $\hat{X}_s = g_s(I_s)$. The design problem of a multiple description system is to find encoders f_s and decoders g_0 and g_s that minimize the central distortion $E[d(X, \hat{X}_0)]$, with given constraints on side distortions $E[d(X, \hat{X}_s)] < D_s$, for all combinations of rates R_s . The region of interest in $(R_1, R_2, D_0, D_1, D_2)$ is the performance region, which consists of all achievable combinations.

Bounds on MDC Performance

For the single channel case, the rate-distortion theorem in rate-distortion theory, e.g., [15.39], provides the achievable rates for a given distortion. For the case of multiple descriptions with the given bounds D_s on each side distortion, the achievable rates for the side descriptions are

$$R_s \geq I(X; \hat{X}_s), \quad (15.5)$$

for each channel $s \in \{1, 2\}$, where $I(X; \hat{X})$ is the mutual information between X and \hat{X} .

For the central description, when both channels are in the working state, the total rate is bounded as

$$R_1 + R_2 \geq I(X; \hat{X}_1, \hat{X}_2) + I(\hat{X}_1; \hat{X}_2), \quad (15.6)$$

according to *El Gamal and Cover* in [15.40]. Unfortunately, this bound is usually loose.

Interpretation of the bounded multiple description region is straightforward. Any mutual information between the side descriptions increases the total rate of the system. If the design is such that no mutual information exists between the two side descriptions, then

$$R_1 + R_2 \geq I(X; \hat{X}_1, \hat{X}_2) = I(X; \hat{X}_1) + I(X; \hat{X}_2), \quad (15.7)$$

and the effective rate is maximized. In this case, all redundancy is lost, which leads to the minimum possible central distortion $D_0 = D(R_1 + R_2)$.

In the other extreme, maximizing redundancy gives two identical descriptions and

$$R_1 + R_2 \geq 2I(X; \hat{X}_s). \quad (15.8)$$

However, this increases the central distortion to $D_0 = D(R_s) = D_s$ and nothing is gained in terms of distortion. Note that this is the same setup as the simple FEC method in the beginning of the section on Media-Dependent FEC in Sect. 15.4.1.

Bounds for Gaussian Sources

The bounds of the multiple description region are, as stated earlier, generally loose and the region is not fully known. Only for the case of memoryless Gaussian sources and the squared error distortion criterion is the region fully known. While this distribution is not representative of the speech itself, the result provides insight to the performance of multiple description coding. *Ozarow* showed in [15.41] that the region defined by the bounds of El Gamal and Cover are tight in this case and define all achievable combinations of rates and distortions. The bounds on distortions as functions of rate are

$$D_s \geq \sigma^2 2^{-2R_s} \quad (15.9)$$

$$D_0 \geq \sigma^2 2^{-2(R_1+R_2)} \cdot \gamma_D(R_1, R_2, D_1, D_2), \quad (15.10)$$

where the variance σ^2 is used and

$$\gamma_D = \begin{cases} 1 & \text{if } D_1 + D_2 > \sigma^2 + D_0, \\ \frac{1}{1-a^2} & \text{otherwise,} \end{cases} \quad (15.11)$$

$$a = \sqrt{(1-D_1)(1-D_2)} - \sqrt{D_1 D_2 - 2^{-2(R_1+R_2)}}. \quad (15.12)$$

Interpretation of the bounds on side distortion is trivial. The central distortion is increased by a factor of γ_D for low side distortions. This implies that the best achievable distortion for rate $R_1 + R_2$, i. e., $D(R_1 + R_2)$, is obtained

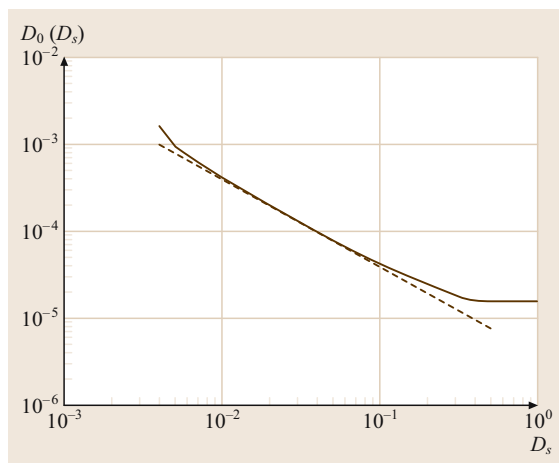


Fig. 15.13 Central distortion D_0 as a function of the side distortion D_s for a Gaussian i.i.d. source with unit variance and fixed per channel rate $R = 4$. Note the trade-off between side and central distortion. The *dashed line* represents the high-rate approximation

only if side distortions are allowed to be large. The relationship is illustrated in a plot of D_0 as a function of $D_s = D_1 = D_2$ for a fixed $R = R_1 = R_2$, in Fig. 15.13.

A high rate approximation to the bound derived by Ozarow is derived in [15.42] and is for the unit-variance Gaussian case given by

$$D_0 D_s = \frac{1}{4} 2^{-4R}. \quad (15.13)$$

MDC Versus Channel Splitting

In modern MDC methods, the side distortions decrease with increasing rate. A technique relevant to speech that does not have this property but is generally included in discussions of MDC is channel splitting by odd-even separation. In this approach odd and even channels are transmitted in separate packets, and if a packet is lost interpolation (in some cases with the aid of an additional coarse quantizer) is used to counter the effect of the loss. Variants on this approach are found in [15.43–46] and a discussion on early unpublished work on this topic can be found in [15.47]. The method is perhaps more accurately classified as a Wyner–Ziv-type coding method (or distributed source coding method) [15.48, 49] with the odd and the even samples forming correlated sources that are encoded separately. In general, the performance of the odd–even separation based methods are suboptimal. The reason is that the redundancy between descriptions is highly dependent of the redundancy present in the speech signal. Hence, the trade-off between the side distortion and the central distortion is difficult to control.

MDC Scalar Quantization

Two methods for design of multiple description scalar quantization, resolution-constrained and entropy-constrained, were described by Vaishampayan in [15.50] and [15.51]. In resolution-constrained quantization, the codeword length is fixed and its resolution is constrained. In entropy-constrained quantization the codeword length is variable and the index entropy is constrained. Vaishampayan assumes the source to be represented by a stationary and ergodic random process X . The design of the resolution-constrained coder is described next, followed by a description of the entropy-constrained coder.

The encoder first maps a source sample x to a partition cell in the central partition $\mathcal{A} = \{A_1, \dots, A_N\}$. The resulting cell with index i_0 in \mathcal{A} is then assigned two indices via the index assignment function $a(i_0) = \{i_1, i_2\}$, where $i_s \in \mathcal{I}_s = \{1, 2, \dots, M_s\}$ and $N \leq M_1 M_2$. The

index assignment function is such that an inverse $i_0 = a^{-1}(i_1, i_2)$ referring to cell A_{i_0} always exists. The two indices, resolution-constrained, are sent over each channel to the receiver. As is appropriate for constrained-resolution coding, the codewords are of equal length.

At the receiver end, depending on which channels are in a working state, a decoder is engaged. In the case when one of the channels is not functioning, the received index of the other channel is decoded by itself to a value $\hat{x}_s = g_s(i_s)$ in the corresponding codebook $\hat{\mathcal{X}}_s = \{\hat{x}_{s,1}, \hat{x}_{s,2}, \dots, \hat{x}_{s,M_s}\}$. In the case when both channels deliver an index, these are used to form the central reconstruction value $\hat{x}_0 = g_0(i_1, i_2)$ in codebook $\hat{\mathcal{X}}_0 = \{\hat{x}_{0,1}, \hat{x}_{0,2}, \dots, \hat{x}_{0,N}\}$. The three decoders are referred to as $\mathbf{g} = \{g_0, g_1, g_2\}$.

The performance of the outlined coder is evaluated in terms of a Lagrangian, $L(\mathcal{A}, \mathbf{g}, \lambda_1, \lambda_2)$, which is dependent on the choice of partition \mathcal{A} , decoders \mathbf{g} , and Lagrange multipliers λ_1 and λ_2 . The Lagrange multipliers weight the side distortions. The codeword length constraints are implicit in the partition definition and do not appear explicitly in the Lagrangian. Minimizing the Lagrangian L is done with a training algorithm that is based on finding non-increasing values of L by iteratively holding \mathcal{A} and \mathbf{g} constant while optimizing for \mathbf{g} and \mathcal{A} , respectively.

The performance of the training algorithm presented is highly dependent on the index assignment $a(i_0) = \{i_1, i_2\}$. This mapping should be such that it minimizes the distortion for given channel rates $R_s = \log_2(M_s)$. The minimization is done by comparing all possible combinations of indices i_1 and i_2 for all possible cardinalities of \mathcal{A} , $N \leq M_1 M_2$. This kind of search is too complex, as the total number of possible index assignments is $\sum_{N=1}^{M_1 M_2} (M_1 M_2)! / (M_1 M_2 - N)!$. A suboptimal search algorithm is presented in [15.52]. An illustration of an index assignment that is believed

	1	2	3	4	5	6	7	8
1	1	3						
5	2	4	5					
3		6	7	9				
4			8	10	11			
5				12	13	15		
6					14	16	17	
7						18	19	21
8							20	22

Fig. 15.14 Nested index assignment matrix for $R_s = 3$ bits. Vertical and horizontal axis represent the first and second channel, respectively

to be close to optimal, the *nested index assignment* [15.50], is found in Fig. 15.14. There, the index assignment matrix shows the procedure of inverse mapping $i_0 = a^{-1}(i_1, i_2)$. It is clear that the redundancy in the system is directly dependent on N . The more cells the central partition has, the lower central distortion is achievable at the expense of side distortions.

The performance of resolution-constrained multiple description scalar quantization was analyzed for high rates in [15.42]. It was shown that the central and side distortions for a squared distortion criterion are dependent of the source probability distribution function $p(x)$ as

$$D_0 = \frac{1}{48} \left(\int_{-\infty}^{\infty} p^{1/3}(x) dx \right)^3 2^{-2R(1+a)} \quad (15.14)$$

$$D_s = \frac{1}{12} \left(\int_{-\infty}^{\infty} p^{1/3}(x) dx \right)^3 2^{-2R(1-a)}, \quad (15.15)$$

where $a \in (0, 1)$. Writing this interdependency as the product of the distortions gives better understanding, since the equation then is independent of a . For the special case of a unit-variance Gaussian source we have

$$D_0 D_s = \frac{1}{4} \left(\frac{2\pi e}{4e^{3^{-1/2}}} \right)^2 2^{-4R}. \quad (15.16)$$

The gap of this high-rate scalar resolution-constrained quantizer to the rate-distortion bound computed by Ozarow is 8.69 dB.

The design of the entropy-constrained coder resembles the resolution-constrained coder described above. The differences are the replacement of fixed codeword length constraints by entropy constraints on index entropies, as well as added variable-length coders prior to transmission over each channel.

The Lagrangian function for the entropy-constrained case, $L(\mathcal{A}, \mathbf{g}, \lambda_1, \lambda_2, \lambda_3, \lambda_4)$, depends on six variables. These are the partition \mathcal{A} , decoders \mathbf{g} and Lagrange multipliers λ_1 through λ_4 , where two of the Lagrangian multipliers correspond to the constraints on the index entropies. Thus, in this case the rate constraints are explicitly in the Lagrangian. The Lagrangian is, as for the resolution-constrained case, minimized with a training algorithm. Before sending indices over each channel, variable length coders are applied to indices obtained by the index assignment.

Disregarding the requirement that codewords should have equal length results in improved performance. The high-rate approximation of the product between central and side distortions for the unit-variance Gaussian case is

$$D_0 D_s = \frac{1}{4} \left(\frac{2\pi e}{12} \right)^2 2^{-4R}. \quad (15.17)$$

The resulting gap of the scalar constrained-entropy quantizer to the rate-distortion bound is here 3.07 dB.

MDC Vector Quantization

It is well known from classical quantization theory that the gap between the rate-distortion bound and the performance of the scalar quantizer can be closed by using vector quantization. The maximum gain of an optimal vector quantizer over an optimal scalar quantizer due to the space filling advantage is 1.53 dB [15.53]. This motivates the same approach in the case of multiple descriptions. One approach to multiple description vector quantization is described in the following. It uses lattice codebooks and is described in [15.54], where implementation details are provided for the A_2 and Z_i lattices, with $i = 1, 2, 4, 8$.

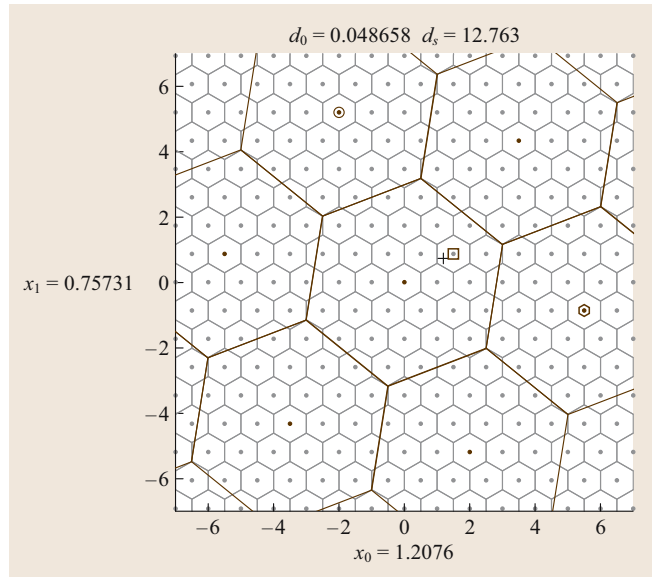


Fig. 15.15 Encoding of the vector marked with a cross. The two descriptions are marked with brown circle and hex. When both descriptions are received the reconstruction point is chosen as the brown square. Central and side distortions are shown at the top

Multiple description vector quantization with lattice codebooks is a method for encoding a source vector $\mathbf{x} = [x_1, \dots, x_n]^T$ with two descriptions. In [15.54], an algorithm for two symmetric descriptions, equal side distortions, is described. These two descriptions are, as with multiple description scalar quantization, sent over different channels, providing a minimum fidelity in the case of failure of one of the channels. Figure 15.15 illustrates the encoded and decoded codewords for a two-dimensional source vector \mathbf{x} . The finer lattice Λ represents the best resolution that can be obtained, that is with the central reconstruction value. Codewords that are sent over the channel are actually not a part of the fine lattice Λ , but of a geometrically similar sublattice Λ' . The sublattice Λ' is obtained by scaling, rotating, and possibly reflecting Λ . This procedure is dependent on what trade-off one wants to obtain between central and side distortions. Which points in Λ' to send and how to choose a reconstruction point in Λ given these points is called the labeling problem, which is comparable to the index assignment problem in the one dimensional case. The labeling problem is solved by setting up a mapping $\Lambda \rightarrow \Lambda' \times \Lambda'$ for all cells of Λ that are contained within the central cell of Λ' . This mapping is then extended to all cells in Λ using the symmetry of the lattices.

Choosing distortions is done by scaling of the used lattices. However, the constraint that the side-distortions must be equal is a drawback of the method. This is improved in [15.55], where an asymmetric multiple description lattice vector quantizer is presented.

High-rate approximations for the setup described above with coding of infinitely large vectors result in a product of distortions given by

$$D_0 D_s = \frac{1}{4} \left(\frac{2\pi e}{2\pi e} \right)^2 2^{-4R} \quad (15.18)$$

for a unit-variance Gaussian source.

Looking closer at the equation, the gain compared to the approximation of the rate-distortion bound is 0dB. This means that the high-rate approximation of the rate-distortion bound has been reached. In an implementation, however, finite-dimension vectors must be used. For the example provided above, with the hexagonal A_2 lattice, the distortion product is

$$D_0 D_s = \frac{1}{4} \left(\frac{2\pi e}{5/36\sqrt{3}} \right)^2 2^{-4R} \quad (15.19)$$

and a gap of 1.53 dB remains compared to the approximation of the rate distortion bound. Using higher dimensions results in greater gains.

Correlating Transforms

Multiple description coding with correlating transforms was first introduced by Wang, Orchar, and Reibman in [15.56]. Their approach to the MDC problem is based on a linear transformation of two random variables, which introduces correlation in the transform coefficients. This approach was later generalized by Goyal and Kovacevic in [15.57] to M descriptions of an N -dimensional vector, where $M \leq N$.

Consider a vector $\mathbf{y} = [y_1, y_2]^T$ with variances σ_1 and σ_2 . Transforming the vector with an orthogonal transformation such as

$$\mathbf{z} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \mathbf{y}$$

produces transform coefficients $\mathbf{z} = [z_1, z_2]^T$. Sending one transform coefficient over each channel results in central distortion

$$D_0 = \frac{\pi e}{6} \left(\frac{\sigma_1^2 + \sigma_2^2}{2} \right) 2^{-2R} \quad (15.20)$$

and average side distortion

$$D_s = \frac{\sigma_1^2 \sigma_2^2}{\sigma_1^2 + \sigma_2^2} + \frac{\pi e}{12} \left(\frac{\sigma_1^2 + \sigma_2^2}{2} \right) 2^{-2R} \quad (15.21)$$

on vector \mathbf{y} . When interpreting the results, we see that the relation of σ_1 and σ_2 determines the trade-off between central and side distortion. If σ_1 equals σ_2 , the distortions obtained are equal to what would have been obtained if \mathbf{y} was sent directly, i. e., the case with no redundancy between descriptions.

The example above leads to the conclusion that a transformation on the i.i.d. source vector \mathbf{x} needs to be performed to get transform coefficients \mathbf{y} of different variances. These are then handled as described above.

Extension to an N -dimensional source vector \mathbf{x} is handled in the following manner. A square correlating transform produces N coefficients in \mathbf{y} with varying variances. These coefficients are quantized and transformed to N transform coefficients, which in turn are placed into M sets in an a priori fixed manner. These M sets form the M descriptions. Finally, entropy coding is applied to the descriptions.

High-rate approximations of multiple description coding with correlating transforms does not tell us anything about the decay of distortion. This is so since

independent of choice of transform $D_0 = O(2^{-2R})$ and $D_1 = D_2 = O(1)$. Simulations [15.57] show however that multiple description coding with correlating transforms performs well at low redundancies.

Conclusions on MDC

This section has presented the bounds on achievable rate-distortion for multiple description coding and a number of specific algorithms have been discussed. We provided some useful comparisons.

Figure 15.16 shows an overview of the performance of the discussed algorithms for i.i.d. Gaussian signals and the squared error criterion. The figure is based on the high-rate approximations of achievable side and central distortions of resolution-constrained multiple description scalar quantization, entropy-constrained multiple description scalar quantization and multiple description vector quantization with dimension two. It is clear that the distortions achievable decrease in the order that the methods are mentioned. Using vectors of dimension larger than two reduces the distortions further, approaching the high-rate approximation of the Ozarow bound for vectors of infinite dimension. Note that multiple description with correlating transforms is not included in the figure, due to the reasons mentioned in the previous section.

Evaluating the performance of the three methods proposed by Vaishampayan [15.50, 51, 54], i.e., constrained-resolution MDC scalar quantization, constrained-entropy MDC scalar quantization and MDC vector quantization with lattice codebooks for con-

strained-entropy coding, the results are not surprising. They are consistent with what is known in equivalent single description implementations. Entropy-constrained quantization performs better than resolution-constrained quantization because the average rate constraint is less stringent than the fixed rate constraint. Even for the i.i.d. case, vector quantization outperforms scalar quantization because of the space filling and shape (constrained resolution only) advantages [15.58].

Improved performance has a price. While the quantization step is often of lower computational complexity for entropy-constrained quantization, it requires an additional lossless encoding and decoding step. The drawbacks of vector quantization are that it introduces delay in the signal and increases computational effort. For constrained-entropy quantizers the added cost generally resides in the variable-length encoding of the quantization indices. For constrained-resolution coding the codebook search procedure generally increases rapidly in computational effort with increasing vector dimension.

The usage of correlating transforms to provide multiple descriptions has been shown to work well for low redundancy rates [15.57]. However, better performance is obtained with the approaches using an index assignment when more redundancy is required. Hence, one might choose not to implement correlating transforms even for low-redundancy applications to avoid changes in the coder design if additional redundancy is required at a later time.

When it comes to speech applications, the described multiple description coding techniques are in general not directly applicable. The speech source is not i.i.d., which was the basic assumption in this section. Thus, the assumption that speech is an i.i.d. source is associated with performance loss. Only pulse-code modulation (PCM) systems are based on the i.i.d. assumption, or perhaps more correctly, ignore the memory that is existent in speech. It is straightforward to adapt the PCM coding systems to using the described multiple description theory, and a practical example is [15.59]. Other speech coding systems are generally based on prediction [e.g., differential PCM (DPCM), adaptive DPCM (ADPCM), and code excited linear prediction (CELP)] making the application of MDC less straightforward. The MDC theory can be applied to the encoding of the prediction parameters (if they are transmitted as side information) and to the excitation. However, the prediction loops at encoder and decoder are prone to mismatch and, hence, error propagation. An alternative approach to exploiting the memory of the source has

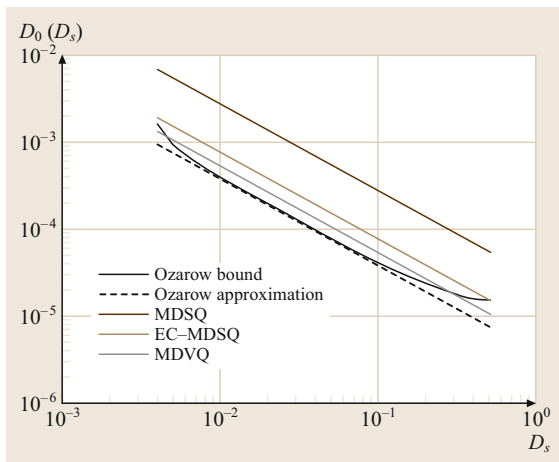


Fig. 15.16 A comparison of the Ozarow bound, its high-rate approximation, and the high-rate approximations of three coders

been described in [15.60]. A Karhunen–Loève transform is applied to decorrelate a source vector prior to usage of MDC. However, since transform coding is not commonly used in speech coders, this method is not applicable either.

15.5 Packet Loss Concealment

If no redundancy is added at the encoder and all processing to handle packet loss is performed at the decoder the approach is often referred to as packet loss concealment (PLC), or more generally error concealment. Sometimes, a robust encoding scheme is combined with a PLC, where the latter operates as a safety net, set up to handle lost packets that the former did not succeed in recovering.

Until recently, two simple approaches to dealing with lost packets have prevailed. The first method, referred to as zero stuffing, involves simply replacing a lost packet with a period of silence of the same duration as the lost packet. Naturally, this method does not provide a high-quality output and, already for packet loss rates as low as 1%, very annoying artifacts are apparent. The second method, referred to as packet repetition, assumes that the difference between two consecutive speech frames is quite small and replaces the lost packet by simply repeating the previous packet. In practice, it is virtually impossible to achieve smooth transitions between the packets with this approach. The ear is very sensitive to discontinuities which leads to audible artifacts as a result of the discontinuities. Furthermore, even a minor change in pitch frequency is easily detected by the human ear. However, this approach performs fairly well at low packet loss probabilities (less than 3%).

More-advanced approaches to packet loss design are based on signal analysis where the speech signal is extrapolated or interpolated to produce a more natural-sounding concealment. These approaches can be divided into two basic classes: nonparametric and parametric methods. If the jitter buffer contains one or more future packets in addition to the past packets, the missing signal can be generated by interpolation. Otherwise, the waveform in the previous frame is extrapolated into the missing frame. Two-sided interpolation generally gives better results than extrapolation.

15.5.1 Nonparametric Concealment

Overlap-and-add (OLA) techniques, originally developed for time-scale modification, belong to the class of

Multiple description theory has made large advances in the last two decades. The results in artificial settings are good. However, much work remains to be done before its promise is fulfilled in practical speech coding applications.

nonparametric concealment methods used in VoIP. The missing frame is in OLA generated by time-stretching the signal in the adjacent frames. The steps of the basic OLA [15.61] are:

1. Extract regularly spaced windowed speech segments around sampling instants $\tau(kS)$.
2. Space windowed segments according to time scaling (regularly spaced S samples apart).
3. Normalize the sum of segments as

$$y(n) = \frac{\sum_k v(n - kS)x(n + \tau(kS) - kS)}{\sum_k v(n - kS)}, \quad (15.22)$$

where $v(n)$ is a window function. Due to the regular spacing and with a proper choice of symmetric window the denominator becomes constant, i. e., $\sum_k v(n - kS) = C$, and the synthesis is thus simplified.

OLA adds uncorrelated segments with similar short-term correlations, thus retaining short-term correlations. However, the pitch structure, i. e., spectral fine structure, has correlations that are long compared to the extraction window and these correlations are destroyed, resulting in poor performance. Two significant improvement approaches are synchronized OLA (SOLA) [15.62] and waveform similarity OLA (WSOLA) [15.63].

In SOLA the extracted windowed speech segments are regularly spaced as in OLA, but when spacing the output segments they are repositioned such that they have high correlation with the already formed portion. The generated segment is formed as

$$y(n) = \frac{\sum_k v(n - kS + \delta_k)x(n + \tau(kS) - kS + \delta_k)}{\sum_k v(n - kS + \delta_k)}. \quad (15.23)$$

A renormalization is needed after placing the segment due to the nonconstant denominator. The window shift δ_k is searched and selected such that the cross-correlation

between the windowed segment $v(n - kS + \delta_k)x(n + \tau(kS) - kS + \delta_k)$ and the previously generated output

$$y_{k-1}(n) = \frac{\sum_{m=-\infty}^{k-1} v(n - mS + \delta_m)x(n + \tau(mS) - mS + \delta_m)}{\sum_{m=-\infty}^{k-1} v(n - mS + \delta_m)} \quad (15.24)$$

is maximized.

WSOLA, instead, extracts windowed segments that are selected for maximum cross-correlation with the last played out segment and regularly space these at the output. The constant denominator implies that there is no need to re-normalize and **WSOLA** is thus simpler than **SOLA**.

$$y(n) = \frac{\sum_k v(n - kS)x(n + \tau(kS) - kS + \delta_k)}{\sum_k v(n - kS)} = \sum_k v(n - kS)x(n + \tau(kS) - kS + \delta_k). \quad (15.25)$$

An example of **PLC** using **WSOLA** is presented in [15.64].

The two techniques can efficiently be utilized in compensating for packet delays, as well as for pure **PLC** as mentioned in Sect. 15.3.5. For example, if a packet is not lost and only delayed less than a full frame interval the signal can be stretched this time amount until the play-out of the delayed frame starts. Examples of this are [15.25] for **SOLA** and [15.24] for **WSOLA**.

15.5.2 Parametric Concealment

Waveform substitution methods were early **PLC** methods that tried to find a pitch cycle waveform in the previous frames and repeat it. *Goodman* et al. [15.65] introduced two approaches to waveform substitution,

a pattern matching approach and a pitch detection approach. The pattern matching approach used the last samples of the previous frame as a template and searched for a matching segment earlier in the received signal. The segment following this match was then used to generate the missing frame. The pitch detection approach estimated the pitch period and repeated the last pitch cycle. According to [15.66], the pitch detection approach yielded better results.

Voicing, power spectrum and energy are other example of features, besides the pitch period, that can be estimated for the previous speech segments and extrapolated in the concealed segment. The packet loss concealment method for the waveform codec G.711, as specified in annex I [15.67] to the **ITU** standard, is an example of such a method.

If the **PLC** is designed for a specific codec and has access to the decoder and its internal states, better performance can generally be obtained than if only the decoded waveform is available. Many codecs, such as G.729, have a built-in packet loss concealment algorithm that is based on knowledge of the decoder parameters used for extrapolation.

Recent approaches to **PLC** include different types of speech modeling such as linear prediction [15.68], sinusoidal extrapolation [15.69], multiband excitation [15.70], and nonlinear oscillator models [15.71]. The latter uses an oscillator model of the speech signal, consisting of a transition codebook built from the existing signal, which predicts future samples. It can advantageously be used for adaptive play-out similar to the **OLA** methods. Further, in [15.72], a **PLC** method is presented for **LPC**-based coders, where the parameter evolution in the missing frames is determined by hidden Markov models.

Concealment systems that are not constrained by producing an integer number of whole frames, thus becoming more flexible in the play-out, have the potential to produce higher-quality recovered speech. This has been confirmed by practical implementations [15.23].

15.6 Conclusion

Designing a speech transmission system for **VoIP** imposes many technical challenges, some similar to those of traditional telecommunications design, and some specific to **VoIP**. The most important challenges for **VoIP** are a direct result of the characteristics of the transport media — **IP** networks. We showed that overall delay of such networks can be a problem for **VoIP** and that,

particularly for the small packets common in **VoIP**, significant packet loss often occurs at network loads that are far below the nominal capacity of the network. It can be concluded that, if a **VoIP** system is to provide the end user with low transmission delay and with high voice quality, it must be able to handle packet loss and transmission time jitter efficiently.

In this chapter, we discussed a number of techniques to address the technical challenges imposed by the network on VoIP. We concluded that, with proper design, it is generally possible to achieve VoIP voice quality that is equal or even better than PSTN. The extension of the signal bandwidth to 8 kHz is a major contribution towards improved speech quality. Multiple description coding is a powerful technique to address packet loss and delay in an efficient manner. It has been

proven in practical applications and provides a theoretical framework that facilitates further improvement. Significant contributions towards robustness and minimizing overall delay can also be made by the usage of adaptive jitter buffers that provide flexible packet loss concealment. The combination of wide-band, multiple description coding, and packet loss concealment facilitates VoIP with high speech quality and a reasonable latency.

References

- 15.1 L.R. Rabiner, R.W. Schafer: *Digital Processing of Speech Signals* (Prentice Hall, Englewood Cliffs 1978)
- 15.2 W. Stallings: *High-Speed Networks: TCP/IP and ATM Design Principles* (Prentice Hall, Englewood Cliffs 1998)
- 15.3 Information Sciences Institute: Transmission control protocol, IETF **RFC793** (1981)
- 15.4 J. Postel: User datagram protocol, IETF **RFC768** (1980)
- 15.5 H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson: RTP a transport protocol for real-time applications, IETF **RFC3550** (2003)
- 15.6 ITU-T: G.131: Talker echo and its control (2003)
- 15.7 ITU-T: G.114: One-way transmission time (2003)
- 15.8 C.G. Davis: An experimental pulse code modulation system for short haul trunks, *Bell Syst. Tech. J.* **41**, 25–97 (1962)
- 15.9 IEEE: 802.11: Part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications (2003)
- 15.10 IEEE: 802.15.1: Part 15.1: Wireless medium access control (MAC) and physical layer (PHY) specifications for wireless personal area networks (WPANs) (2005)
- 15.11 E. Dimitriou, P. Sörqvist: Internet telephony over WLANs, 2003 USTAs Telecom Eng. Conf. Supercomm (2003)
- 15.12 ITU-T: G.711: Pulse code modulation (PCM) of voice frequencies (1988)
- 15.13 IEEE: 802.1D Media access control (MAC) bridges (2004)
- 15.14 D. Grossman: New terminology and clarifications for diffserv, IETF **RFC3260** (2002)
- 15.15 R. Braden, L. Zhang, S. Berson, S. Herzog, S. Jamin: Resource ReSerVation Protocol (RSVP) – Version 1 Functional specification, IETF **RFC2205** (1997)
- 15.16 C. Aurrecochea, A.T. Campbell, L. Hauw: A survey of QoS architectures, *Multimedia Syst.* **6**(3), 138–151 (1998)
- 15.17 IEEE: 802.11e: Medium Access Control (MAC) Quality of Service (QoS) Enhancements (2005)
- 15.18 E. Hänsler, G. Schmidt: *Acoustic Echo and Noise Control – A Practical Approach* (Wiley, New York 2004)
- 15.19 ITU-T: G.729: Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear prediction (CS-ACELP) (1996)
- 15.20 S. Andersen, A. Duric, H. Astrom, R. Hagen, W.B. Kleijn, J. Linden: Internet Low Bit Rate Codec (iLBC), IETF **RFC3951** (2004)
- 15.21 Ajay Bakre: www.globalipsound.com/datasheets/isac.pdf (2006)
- 15.22 S.B. Moon, J.F. Kurose, D.F. Towsley: Packet audio playout delay adjustment: Performance bounds and algorithms, *Multimedia Syst.* **6**(1), 17–28 (1998)
- 15.23 Ajay Bakre: www.globalipsound.com/datasheets/neteq.pdf (2006)
- 15.24 Y. Liang, N. Farber, B. Girod: Adaptive playout scheduling and loss concealment for voice communication over IP networks, *IEEE Trans. Multimedia* **5**(4), 257–259 (2003)
- 15.25 F. Liu, J. Kim, C.-C.J. Kuo: Adaptive delay concealment for internet voice applications with packet-based time-scale modification, *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.* (2001)
- 15.26 ITU-T: P.800: Methods for subjective determination of transmission quality (1996)
- 15.27 S. Pennock: Accuracy of the perceptual evaluation of speech quality (PESQ) algorithm, *Proc. Measurement of Speech and Audio Quality in Networks* (2002)
- 15.28 M. Varela, I. Marsh, B. Grönvall: A systematic study of PESQs behavior (from a networking perspective), *Proc. Measurement of Speech and Audio Quality in Networks* (2006)
- 15.29 ITU-T: 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 (2001)
- 15.30 ITU-T: P.862.1: Mapping function for transforming P.862 raw result scores to MOS-LQ0 (2003)
- 15.31 C. Perkins, O. Hodson, V. Hardman: A survey of packet loss recovery techniques for streaming audio, *IEEE Network* **12**, 40–48 (1998)
- 15.32 J. Rosenberg, H. Schulzrinne: An RTP payload format for generic forward error correction, IETF **RFC2733** (1999)

- 15.33 J. Lacan, V. Roca, J. Peltotalo, S. Peltotalo: Reed–Solomon forward error correction (FEC), IETF (2007), work in progress
- 15.34 J. Rosenberg, L. Qiu, H. Schulzrinne: Integrating packet FEC into adaptive voice playout buffer algorithms on the internet, Proc. Conf. Comp. Comm. (IEEE INFOCOM 2000) (2000) pp. 1705–1714
- 15.35 W. Jiang, H. Schulzrinne: Comparison and optimization of packet loss repair methods on VoIP perceived quality under bursty loss, Proc. Int. Workshop on Network and Operating System Support for Digital Audio and Video (2002)
- 15.36 E. Martinian, C.-E.W. Sundberg: Burst erasure correction codes with low decoding delay, IEEE Trans. Inform. Theory **50**(10), 2494–2502 (2004)
- 15.37 C. Perkins, I. Kouvelas, O. Hodson, V. Hardman, M. Handley, J. Bolot, A. Vega-Garcia, S. Fosse-Parisis: RTP payload format for redundant audio data, IETF **RFC2198** (1997)
- 15.38 J.-C. Bolot, S. Fosse-Parisis, D. Towsley: Adaptive FEC-based error control for internet telephony, Proc. Conf. Comp. Comm. (IEEE INFOCOMM '99) (IEEE, New York 1999) p. 1453–1460
- 15.39 T.M. Cover, J.A. Thomas: *Elements of Information Theory* (Wiley, New York 1991)
- 15.40 A.A.E. Gamal, T.M. Cover: Achievable rates for multiple descriptions, IEEE Trans. Inform. Theory **IT-28**(1), 851–857 (1982)
- 15.41 L. Ozarow: On a source coding problem with two channels and three receivers, Bell Syst. Tech. J. **59**, 1909–1921 (1980)
- 15.42 V.A. Vaishampayan, J. Batllo: Asymptotic analysis of multiple description quantizers, IEEE Trans. Inform. Theory **44**(1), 278–284 (1998)
- 15.43 N.S. Jayant, S.W. Christensen: Effects of packet losses in waveform coded speech and improvements due to an odd–even sample–interpolation procedure, IEEE Trans. Commun. **COM-29**(2), 101–109 (1981)
- 15.44 N.S. Jayant: Subsampling of a DPCM speech channel to provide two self-contained half-rate channels, Bell Syst. Tech. J. **60**(4), 501–509 (1981)
- 15.45 A. Ingle, V.A. Vaishampayan: DPCM system design for diversity systems with applications to packetized speech, IEEE Trans. Speech Audio Process. **3**(1), 48–58 (1995)
- 15.46 A.O.W. Jiang: Multiple description speech coding for robust communication over lossy packet networks, IEEE Int. Conf. Multimedia and Expo (2000) pp. 444–447
- 15.47 V.K. Goyal: Multiple description coding: Compression meets the network, IEEE Signal Process. Mag. **18**, 74–93 (2001)
- 15.48 A.D. Wyner: Recent results in the Shannon theory, IEEE Trans. Inform. Theory **20**(1), 2–10 (1974)
- 15.49 A.D. Wyner, J. Ziv: The rate–distortion function for source coding with side information at the decoder, IEEE Trans. Inform. Theory **22**(1), 1–10 (1976)
- 15.50 V.A. Vaishampayan: Design of multiple description scalar quantizers, IEEE Trans. Inform. Theory **IT-39**(4), 821–834 (1993)
- 15.51 V.A. Vaishampayan, J. Domaszewicz: Design of entropy-constrained multiple-description scalar quantizers, IEEE Trans. Inform. Theory **IT-40**(4), 245–250 (1994)
- 15.52 N. Görtz, P. Leelapornchai: Optimization of the index assignments for multiple description vector quantizers, IEEE Trans. Commun. **51**(3), 336–340 (2003)
- 15.53 R.M. Gray: *Source Coding Theory* (Kluwer, Dordrecht 1990)
- 15.54 V.A. Vaishampayan, N.J.A. Sloane, S.D. Servetto: Multiple-description vector quantization with lattice codebooks: Design and analysis, IEEE Trans. Inform. Theory **47**(1), 1718–1734 (2001)
- 15.55 S.N. Diggavi, N. Sloane, V.A. Vaishampayan: Asymmetric multiple description lattice vector quantizers, IEEE Trans. Inform. Theory **48**(1), 174–191 (2002)
- 15.56 Y. Wang, M.T. Orchard, A.R. Reibman: Multiple description image coding for noisy channels by pairing transform coefficients, IEEE Workshop on Multimedia Signal Processing (1997) pp. 419–424
- 15.57 V.K. Goyal, J. Kovacevic: Generalized multiple description coding with correlating transforms, IEEE Trans. Inform. Theory **47**(6), 2199–2224 (2001)
- 15.58 T. Lookabough, R. Gray: High-resolution theory and the vector quantizer advantage, IEEE Trans. Inform. Theory **IT-35**(5), 1020–1033 (1989)
- 15.59 Ajay Bakre: www.globalipsound.com/datasheets/ipcm-wb.pdf (2006)
- 15.60 J. Batllo, V.A. Vaishampayan: Asymptotic performance of multiple description transform codes, IEEE Trans. Inform. Theory **43**(1), 703–707 (1997)
- 15.61 D.W. Griffin, J.S. Lim: Signal estimation from modified short-time Fourier transform, IEEE Trans. Acoust. Speech Signal Process. **32**, 236–243 (1984)
- 15.62 S. Roucos, A. Wilgus: High quality time-scale modification for speech, Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (1985) pp. 493–496
- 15.63 W. Verhelst, M. Roelands: An overlap-add technique based on waveform similarity (WSOLA) for high quality time-scale modification of speech, Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (1993) pp. 554–557
- 15.64 H. Sanneck, A. Stenger, K. Ben Younes, B. Girod: A new technique for audio packet loss concealment, Proc. Global Telecomm. Conf. GLOBECOM (1996) pp. 48–52
- 15.65 D.J. Goodman, G.B. Lockhart, O.J. Wasem, W.C. Wong: Waveform substitution techniques for recovering missing speech segments in packet voice communications, IEEE Trans. Acoust. Speech Signal Process. **34**, 1440–1448 (1986)
- 15.66 O.J. Wasem, D.J. Goodman, C.A. Dvorak, H.G. Page: The effect of waveform substitution on the qual-

- ity of PCM packet communications, IEEE Trans. Acoust. Speech Signal Process. **36**(3), 342–348 (1988)
- 15.67 ITU-T: G.711 Appendix I: A high quality low-complexity algorithm for packet loss concealment with G.711 (1999)
- 15.68 E. Gündüzhan, K. Momtahan: A linear prediction based packet loss concealment algorithm for PCM coded speech, IEEE Trans. Acoust. Speech Signal Process. **9**(8), 778–785 (2001)
- 15.69 J. Lindblom, P. Hedelin: Packet loss concealment based on sinusoidal extrapolation, Proc. IEEE Int. Conf. Acoust. Speech Signal Process., Vol.1 (2002) pp.173–176
- 15.70 K. Clüver, P. Noll: Reconstruction of missing speech frames using sub-band excitation, Int. Symp. Time-Frequency and Time-Scale Analysis (1996) pp. 277–280
- 15.71 G. Kubin: Proc. IEEE Int. Conf. Acoust. Speech Signal Process., Vol.1 (1996) pp.267–270
- 15.72 C.A. Rodbro, M.N. Murthi, S.V. Andersen, S.H. Jensen: Hidden Markov Model-based packet loss concealment for voice over IP, IEEE Trans. Speech Audio Process. **14**(5), 1609–1623 (2006)