Dynamically Adding Redundancy for Improved Error Concealment in Packet Voice Coding

Levent Tosun

Department of Electrical & Computer Engineering
McGill University
Montreal, Canada

December 2004

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of Master of Engineering.

Abstract

Data is sent in packets of bits over the Internet. However, packets may not arrive in order or in time for playout. Packet loss is a frequently encountered problem in Voice-over-IP (VoIP) applications. Modern speech coders use past information to decode current packets in order to reach very low bit-rates. Therefore, when a packet is lost, the effect of this packet loss propagates over several subsequent packets.

In this thesis, a new redundancy-based packet-loss-concealment scheme is presented. Many redundancy-based packet-loss-concealment schemes send a fixed amount of extra information about the current packet as part of the subsequent packet, but not every packet is equally important for packet loss concealment. We have developed an algorithm to determine the importance of packets and we propose that extra information should only be sent for the important packets. This provides a lower average bit-rate compared to sending the same amount of extra information for each and every packet. We use a linear prediction (LP) based speech coder (ITU-T G.723.1) as a test platform and we propose that only the excitation parameters should be sent as extra information since LP parameters of a frame can be estimated using the LP parameters of the previous frame. Furthermore, we propose that excitation parameters of an important frame that are sent as redundant information should be used in the reconstruction of the lost waveform — as a consequence, the states of the subsequent frame will also be updated.

Sommaire

L'information est transmise à travers l'Internet sous forme de paquets de bits. Cependant, il se peut que ces paquets n'arrivent pas dans l'ordre ou dans le délai prévu. En effet, la perte des paquets est un problème fréquent pour la transmission de la voix sur IP. Les codeurs de la voix les plus récents utilisent de l'information antérieure pour décoder le paquet courant dans le but d'atteindre des niveaux de débit très bas. Conséquemment, lors de la perte d'un paquet, l'effet de cette perte est reporté sur les paquets suivants.

Dans cette thèse, un nouveau schéma pour la dissimulation de la perte de paquets basé sur la redondance est présenté. Plusieurs de ces schémas envoient une quantité fixe d'information supplémentaire pertinente au paquet courant avec le paquet suivant. Cependant, tous les paquets ne partagent pas la même importance lors de la dissimulation des pertes. Nous avons développé un algorithme pour déterminer l'importance des paquets et proposons l'envoi de l'information supplémentaire seulement dans le cas des paquets qui sont jugés importants. Ceci produit un débit moyen moins élevé que dans le cas où la même quantité d'information supplémentaire est envoyée pour chaque paquet. La platforme de test utilisé est un codeur de la voix basé sur la prédiction linéaire (PL) selon le standard ITU-T G.723.1. Nous proposons qu'il suffise d'envoyer les paramètres d'excitation comme information supplémentaire puisque ceux-ci peuvent être estimés grâce aux paramètres du paquet précédent. De plus, ils doivent être utilisés lors de la reconstruction des paquets perdus et lors de la mise à jour des états du codeur.

Acknowledgments

Words are not enough to express my gratitude to Professor Kabal and to my parents for providing me with the chance to study at McGill University and supporting me through my studies.

I am sincerely grateful to Professor Kabal for providing me with the opportunity to do my master's at McGill University under his supervision. Throughout my studies, Professor Kabal, beyond being a very supportive supervisor, has always been a very understanding person. I learned a lot through his guidance. The experience I gained through his supervision is invaluable to me and I have no doubt that I will always benefit from it in all aspects of my professional life. However, more important than his supervision, his understanding has meant a lot to me.

My parents have always been supportive of any decision that I have made. I will always be thankful to them for morally and financially supporting my decision of coming to Canada to continue my studies. They have always told me that my happiness and accomplishments were enough a gift to them, though they would always be supportive of me regardless. This thesis, as an accomplishment of my master's, is my gift to them.

Special thanks go to Colm for his valuable suggestions while reviewing my thesis. I also want to acknowledge Tania for helping me out with the translation of the abstract.

Finally I want to thank all of my friends and colleagues. I don't think it would be possible to be here and have worked it through without their friendship and support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Communication is an important part of everyday life. As a result of technological improvements in digital communication, new methods that enable people to communicate from a distance are introduced everyday. Regardless of the tools used for communication (tv, radio, internet, fax, etc.), in digital communication first, speech is translated into bits, which is called speech coding. Then the bitstream is transmitted to the desired location by some means (wireless, wired, etc.). Finally the bitstream is retranslated into speech at the receiver. Hence, the aim of speech coders is to represent a speech signal with very few bits without sacrificing intelligibility. Modern speech coders achieve very low bit-rates by taking advantage of redundant information found in speech signals. They rely on the assumption that past sections of speech signals provide information about present sections, therefore they use past sections of a speech signal to code and decode present ones. As long as it is guaranteed that the bitstream arrives unaltered at the destination, the only concern of a good speech coder is to achieve a low bit-rate while keeping the quality high enough so as to retain the requisite level of intelligibility. However, with the recent and growing interest in communication over the Internet, the effect of errors (packet loss) occurring in transmission have become a major concern for speech coders.

1.1 Voice Over Internet Protocol Networks

The growing interest in Voice Over Internet Protocol (VoIP) networks necessitates an increasing amount of work in this area to solve the problems faced in the process. The Internet is a packet switched network for which quality of service is not guaranteed. This

means three things:

1. Unlike the methods employed in other communication means, data is not sent as a bitstream, but in packets of bits.

2. Packets sent for transmission experience variable network delays. Therefore, packets may not arrive in order.

3. Packets may not arrive at all.

Real-time voice transmission over the Internet necessitates a limit on the waiting time for the arrival of a packet for the sake of the quality of the conversation. A buffer is used to hold packets until their scheduled playout times, after which the packets are considered lost. Packet loss is a frequently encountered problem in VoIP applications. There has been considerable research in this field, proposing several different methods to conceal the effect of lost packets. Many effective methods rely on sending extra information. This extra information is the most important information in a packet that is used in the decoding process — the information that is needed to adequately regenerate the waveforms that the lost packets correspond to.

1.2 Thesis Contribution

There are different methods to determine the extra information to be sent, and different methods to determine the ways to use the extra information. There is one thing in common in many of the methods relying on sending redundant information: they send a fixed amount of extra information for each and every packet. However packets are not equally important for packet loss concealment. The focus of this research is to define the extra information to be sent and to determine for which packets to send it, thereby achieving a lower average bit-rate than sending the extra information for every packet.

1.3 Thesis Organization

Chapter 2 discusses different speech coders classified according to the methods they employ — waveform coders, parametric coders and hybrid coders. Waveform coders are the simplest speech coders in that they try to preserve the waveform of a speech signal. In

contrast to waveform coders, parametric coders focus on ways to find parameters to model the synthesis of each segment of a speech signal. Finally, hybrid coders are a combination of waveform and parametric coders — they attempt to find the parameters to model the synthesis of each speech segment while also providing an excitation signal that minimizes the error in some sense to drive this model. Hybrid coders combine the strengths of waveform and parametric coders, therefore many modern coders are hybrid. The basis of many parametric and hybrid coders is linear prediction, which is explained when parametric coders are discussed. Hybrid coders are then explained, with details provided on three particular methods used in hybrid coders — code excited linear prediction (CELP), algebraic code excited linear prediction (ACELP), and multi-pulse maximum likelihood quantization (MP-MLQ). In this thesis we use ITU-T (Telecommunication Standardization Section of International Telecommunication Union) G.723.1 as the test platform. G.723.1 is a widely used hybrid speech coder designed for voice transmission over the Internet. We discuss different aspects of G.723.1 in a different section. Evaluations of test results in this thesis use Perceptual Evaluation of Speech Quality (PESQ), a standard and effective tool used to measure the quality of a degraded speech signal by comparing it with the original one. After explaining the details of PESQ, we finish Chapter 2 by illustrating the performance of G.723.1 in terms of PESQ scores.

In Chapter 3, we briefly discuss packet-loss-concealment schemes. We explain the packet-loss-concealment schemes in two categories: receiver-based schemes and sender-receiver-based schemes, with more focus on the latter since they are better in terms of performance. Receiver-based schemes try to reproduce the speech segment that a lost packet corresponds to by using the previous and subsequent packets or replace it with another waveform, which supposedly will not have a big overall negative effect on the quality of the speech. Sender-receiver-based schemes are those which use the transmitter as well as the receiver for packet loss concealment. We explain three methods briefly in this category: priority-based schemes, redundancy-based schemes and interleaving-based schemes. Priority-based schemes assign priority to the packets according to their importance and assume that the packets will be dropped by a supporting network according to the preassigned priorities. Redundancy-based schemes add redundant information at the transmitter about each packet to either the previous or the next packet, which is then used in the receiver in case of a loss. Since they add extra information, the bit-rate of the coder is increased. Interleaving-based schemes, in contrast to redundancy-based schemes, do not add any extra

information and hence do not increase the bit-rate. In interleaving, the information in a packet is distributed into several packets, so that when a packet is lost, only part of the information in that packet is gone and the lost information can be recovered using the part of the information that was distributed to other packets. In other words, in interleaving, loss is spread over several frames. After we discuss the advantages and disadvantages of these three methods, we talk about the requirements of a good packet-loss-concealment scheme for hybrid coders. A recent method to improve the performance of packet-loss-concealment schemes is then explained — using late frames to improve packet recovery. We end the chapter by explaining the details of the packet-loss-concealment scheme used in G.723.1 and illustrating its performance in terms of PESQ scores.

In the fourth chapter, we give experimental results related to the main focus of this research. We first show that certain packets are much more important than others for packet loss concealment. We then examine the effect of sending extra information for the most important packets for two cases: sending the linear prediction (LP) parameters of the most important packets as redundant information as opposed to sending the excitation parameters. We show that sending LP parameters does not make a big improvement, whereas sending excitation parameters of the most important packets as extra information does improve the quality of packet loss concealment significantly. We conclude that it is not necessary to send extra information for all packets, but only for the most important packets. Furthermore, LP parameters of a frame can be regenerated using the LP parameters of the previous frame; however, excitation parameters of the most important packets cannot be reproduced adequately and hence they should be sent as extra information. We then discuss the methods that can be used to determine if a packet is important or not. We first propose that a reference PESQ score can be defined using the first few packets of a speech signal. Upon defining the reference PESQ score, the importance of each packet can be determined by first considering that packet lost, then finding a PESQ score for this case and finally comparing this PESQ score with the reference PESQ score. Then we propose that the importance of packets can be better determined by observing the excitation signals of consecutive packets. We observe that the excitation signals of the most important packets correspond to a voiced section of speech — they have periodic components with significantly larger peaks in amplitude as compared to those of the excitation signals of the packets preceding them. We observe that excitation signals of the packets preceding the most important ones, on the contrary, correspond to an unvoiced section of speech —

compared to the excitation signals of the most important packets, they resemble random noise. We thus conclude that the most important packets are those corresponding to voiced sections of a speech signal following packets corresponding to unvoiced sections and that observing the excitation signals of consecutive packets is a good method to tell if a packet is important. We propose that the amplitudes of the peaks of two consecutive excitation signals can be compared to determine whether a packet is important.

Chapter 2

Speech Coders

Speech coders aim to use very few bits to represent a speech signal after digitization while maintaining a toll quality [1]. Most of the energy of speech signals is found in the frequency range of 300 Hz to 3400 Hz and the intelligibility of speech signals is mostly determined by components in this frequency range [2]. Therefore, in telecommunication applications, due to bandwidth limitations, although high frequency terms slightly improve intelligibility, speech signals are low pass filtered to obtain the frequency terms in this range. For digital communication applications, following the Nyquist theorem, which states that the sampling frequency must be at least twice the bandwidth of the continuous signal to avoid aliasing, low pass filtered speech signals are sampled at 8 kHz as a common practice. Analog samples are then converted to digital format. At least 8 bits should be used per sample to maintain a satisfactory quality [1]. The common practice, though, is to use 16 bits/sample [1]. This gives a bit-rate of 128 kbit/s. A coder is composed of two main parts; encoder and decoder. The encoder aims to reduce this bit-rate and the decoder aims to recover the original speech.

Speech coders can be categorized in three groups according to the methods they employ to reduce the bit-rate.

1. Waveform Coders
2. Parametric Coders
3. Hybrid Coders

2.1 Waveform Coders

Waveform coders try to preserve the waveform of the speech signal. In this type of coders, after sampling and quantizing the analog signal, samples are coded and sent directly. There are different techniques. The simplest method is pulse code modulation (PCM). In PCM, samples are coded directly. For example, if 16-bit precision is used, among the 2^{16} possible levels, the one that is closest to the sample to be coded is selected. The 16-bit code that corresponds to the selected level is then transmitted to the channel. However, there are better ways than coding the samples directly, such as coding the differences between the consecutive samples. When the differences between consecutive samples are coded as opposed to the samples themselves, the range to be coded decreases. For the same bit precision, error decreases. This technique is called differential PCM (DPCM). Both for PCM and DPCM, levels can be positioned nonuniformly in such a way that there are more levels for amplitude ranges in which the probability of having different amplitudes is higher than some other amplitude ranges. For example, it is less likely to have a lot of different amplitude levels close to the maximum and minimum as compared to the range around the mean. A predetermined amplitude distribution can be used for nonuniform level assignment. However, a better but more complex way to do it is making this decision adaptive — starting with an initial distribution function and modifying it according to the new samples. There are many other methods that are employed by waveform coders. Waveform coders offer very high quality speech at the expense of using very high bit-rates. ITU-T G.711 [3] and ITU-T G.726 [4] are two well-known examples of waveform coders.

2.2 Parametric Coders

Parametric coders are based on the assumption that speech signals can be reproduced using a model, such as a digital filter that models the vocal tract of a speaker [5], which can be represented by some parameters. Following this assumption, instead of trying to preserve the shape of the speech signal by encoding the waveform itself, they encode the parameters. Once the parameters are obtained at the receiver end, the speech signal is created using the same model. Parametric coders achieve very low bit-rates but they cannot provide toll quality. Therefore, parametric coders are only used when there are tight bandwidth requirements [5]. Many parametric coders are based on linear prediction.

2.2.1 Linear Prediction

Linear prediction states that if there is correlation between the samples of a signal, then a given sample value can be estimated using the past samples. If we denote the i^{th} sample with $s[i]$ and its estimate with $\hat{s}[i]$, $\hat{s}[n]$ can be formulated as follows:

$$\hat{s}[n] \approx s[n] \tag{2.1}$$

$$\hat{s}[n] = a_1 s[n-1] + a_2 s[n-2] + \ldots + a_K s[n-K]. \tag{2.2}$$

Here K past samples are used to estimate $s[n]$. The coefficients a_i are called the linear prediction coefficients. This equation can be rewritten as follows:

$$\hat{s}[n] = \sum_{i=1}^{K} a_i s[n-i]. \tag{2.3}$$

The more past samples are used to estimate the current sample, the better is the estimation, hence the smaller is the error. If we denote the error of the estimation of the n^{th} sample with $e[n]$ then using Eqs. (2.2) and (2.3), we can formulate $e[n]$ as

$$\begin{aligned} e[n] &= s[n] - \hat{s}[n] && (2.4)\\ &= s[n] - a_1 s[n-1] - a_2 s[n-2] - \ldots - a_K s[n-K] && (2.5)\\ &= s[n] - \sum_{i=1}^{K} a_i s[n-i]. && (2.6) \end{aligned}$$

Linear prediction is the estimation of the linear prediction coefficients a_i that will minimize the error in some sense and there exists several methods for this purpose. If we take the z-transform of both sides of Eq. (2.6) and modify this equation a little bit we get Eq. (2.11).

$$E(z) = \left(1 - \sum_{i=1}^{K} a_i z^{-i}\right) S(z) \tag{2.7}$$

$$A(z) = 1 - \sum_{i=1}^{K} a_i z^{-i} \tag{2.8}$$

$$H(z) = \frac{1}{A(z)} \tag{2.9}$$

$$E(z)H(z) = S(z) \tag{2.10}$$

$$e[n]\left(\frac{1}{1-\sum_{i=1}^{K} a_i z^{-i}}\right) = s[n]. \tag{2.11}$$

This equation can be interpreted as follows: the current sample $s[n]$ can be obtained as the output of a filter provided that the coefficients a_i and the error signal $e[n]$ satisfying the equation are found. Here, the filter models the vocal tract and the error signal models the excitation signal. The purpose of parametric coders is then to find the parameters that will model the vocal tract and to estimate the excitation signal, since the signal can then be reproduced easily. The reproduction of a speech signal is called speech synthesis. The following figure shows the speech synthesis process in terms of a block diagram (Fig. 2.1).

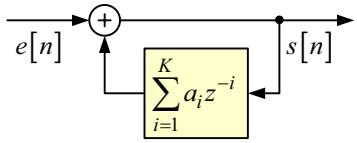


Fig. 2.1 Speech Synthesis

2.2.2 Computation of Linear Prediction Coefficients

We can define a vector to represent the past K samples as follows:

$$\mathbf{s}_K = \begin{bmatrix} s[n-1] \\ s[n-2] \\ \vdots \\ s[n-K] \end{bmatrix}; \tag{2.12}$$

following the same notation, the vector to represent the linear prediction coefficients will be

$$\mathbf{a}_K = \begin{bmatrix} a_1[n] \\ a_2[n] \\ \vdots \\ a_K[n] \end{bmatrix}. \tag{2.13}$$

Then using Eqs. (2.12) and (2.13) we can rewrite Eq. (2.2) as follows:

$$\hat{s}[n] = \mathbf{a}_K^T \mathbf{s}_K. \tag{2.14}$$

Here we used the transpose instead of the Hermitian transpose, because in speech we deal with real values. To minimize the error, one of the methods we can apply is the minimum mean square error (MMSE) method. MMSE, as the name implies, basically means to minimize the mean square error. We want to minimize the mean square error by finding the optimum coefficient vector $\mathbf{a}_K$, thus the mean square is a function of vector $\mathbf{a}_K$. If we denote the mean square error with P then we can formulate P as follows:

$$P(\mathbf{a}_K) = E\{e^2[n]\} = e^2[n], \tag{2.15}$$

which is by using Eq. (2.4) equal to

$$P(\mathbf{a}_K) = (s[n] - \hat{s}[n])^2. \tag{2.16}$$

Using Eq. (2.14) we can rewrite this as follows:

$$\begin{aligned} P(\mathbf{a}_K) &= (s[n] - \mathbf{a}_K^T \mathbf{s}_K)^2 \\ &= s^2[n] + \mathbf{a}_K^T \mathbf{s}_K \mathbf{s}_K^T \mathbf{a}_K - s[n]\mathbf{a}_K^T \mathbf{s}_K - s[n]\mathbf{s}_K^T \mathbf{a}_K. \end{aligned} \tag{2.17}$$

To further simplify this equation we define the following: [6]

$$P_s = s^2[n] \tag{2.18}$$

$$\mathbf{d}_K = \mathbf{s}_K s[n] = \begin{bmatrix} s[n-1]s[n] \\ s[n-2]s[n] \\ \vdots \\ s[n-K]s[n] \end{bmatrix} = \begin{bmatrix} r[-1] \\ r[-2] \\ \vdots \\ r[-K] \end{bmatrix} = \begin{bmatrix} r[1] \\ r[2] \\ \vdots \\ r[K] \end{bmatrix} \tag{2.19}$$

$$\mathbf{R}_K = \mathbf{s}_K \mathbf{s}_K^T = \begin{bmatrix} r[0] & r[1] & \cdots & r[K-1] \\ r[1] & r[0] & \cdots & r[K-2] \\ \vdots & \cdots & \ddots & \vdots \\ r[K-1] & r[K-2] & \cdots & r[0] \end{bmatrix}. \tag{2.20}$$

Here, P_s is the energy of the current sample, which can also be indicated as the desired response since we are trying to estimate it; $\mathbf{d}_K$ is the cross-correlation vector between the current sample and the past K samples to be used to estimate the current sample; and $\mathbf{R}_K$ is the correlation matrix of the vector of the past K samples. The matrix $\mathbf{R}_K$ is guaranteed to be Hermitian and nonnegative definite [6]. Now, using Eqs. (2.18), (2.19) and (2.20) we can rewrite Eq. (2.17) as follows:

$$P(\mathbf{a}_K) = P_s + \mathbf{a}_K^T \mathbf{R}_K \mathbf{a}_K - \mathbf{a}_K^T \mathbf{d}_K - \mathbf{d}_K^T \mathbf{a}_K. \tag{2.21}$$

Now, we have a formula for mean square error and we are trying to find the coefficient vector $\mathbf{a}_K$ that will minimize this. For this purpose we can put this equation into perfect square form as follows:

$$P(\mathbf{a}_K) = P_s - \mathbf{d}_K^T \mathbf{R}_K^{-1} \mathbf{d}_k + (\mathbf{R}_K \mathbf{a}_K - \mathbf{d}_K)^T \mathbf{R}_K^{-1} (\mathbf{R}_K \mathbf{a}_K - \mathbf{d}_K). \tag{2.22}$$

As can be seen, only the third term depends on $\mathbf{a}_K$. Since $\mathbf{R}_K$ is nonnegative definite, $\mathbf{R}_K^{-1}$ is also nonnegative definite. Hence, the minimum value that the third term can get is zero, which means that the minimum value of P we can get is obtained when the third term is equal to zero. When this condition is met, the minimum value of P is found as follows:

$$P(\mathbf{a}_K^o) = P_s - \mathbf{d}_K^T \mathbf{R}_K^{-1} \mathbf{d}_K. \tag{2.23}$$

Here, $\mathbf{a}_K^o$ is the optimizing coefficient vector, in other words, the vector that minimizes the mean square error. MMSE can be obtained when the third term is equal to zero, which is satisfied when

$$\mathbf{R}_K \mathbf{a}_K^o = \mathbf{d}_K. \tag{2.24}$$

Solving this equation to find $\mathbf{a}_K^o$ we can find the MMSE estimator.

To summarize: parametric coders attempt to find parameters that will model speech synthesis and then encode these parameters. Finding LP parameters which will model the vocal tract and the error signal which will model the excitation signal is a common way. This process is called analysis. Figure 2.2 shows the analysis process in terms of a block diagram. The simplified block diagrams of the whole encoding and decoding process are shown in Fig. 2.3.

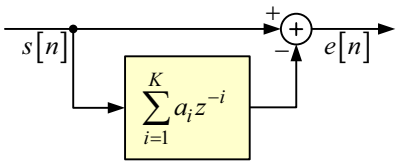


Fig. 2.2 Speech Analysis

(a) Analysis (b) Synthesis

Fig. 2.3 Speech Reproduction: Analysis and Synthesis of Speech

To model a signal with the same parameters, the signal should be stationary. However, speech signals are not stationary — the characteristics change in time. In other words the shape of the vocal tract does not remain the same during speech, so it cannot be modelled by constant parameters. Research shows that the average length of a phoneme is 80 ms [2]. Speech signals are referred to as quasi-stationary because of this reason — the characteristics of the signal remain relatively unchanged for a short period of time. Therefore parametric coders encode speech signals in frames. A typical length of a frame is 20-30 ms [1]. Choosing a frame length in this range means that the characteristics of the signal will remain relatively unchanged during the observation time. In the previous paragraphs, the computation of linear prediction coefficients was discussed. The quasi-stationary nature of speech signals requires LP coefficients to be calculated for each frame. There are efficient recursive algorithms to calculate these parameters. One of them is the Levinson-Durbin Algorithm.

2.2.3 Levinson Durbin Algorithm

The number of past samples used to estimate the current sample is called the order of the estimator. Solving Eq. (2.24) to find the estimator requires finding the inverse of $\mathbf{R}_K$. To find the 10^{th} order estimator, for instance, requires finding the inverse of a 10-by-10 matrix, which is computationally quite complex. In addition to the previous notation, let $\hat{\mathbf{s}}_K$ be the estimate obtained using a K^{th} order estimator $\mathbf{a}_K$. There are recursive algorithms that can efficiently calculate $\mathbf{a}_{K+1}$ and $\hat{\mathbf{s}}_{K+1}$ using $\mathbf{a}_K$ and $\hat{\mathbf{s}}_K$. For this purpose we need to define

some matrices following the notations in [6]:

$$\mathbf{R}_{K+1} = \mathbf{s}_{K+1}\mathbf{s}_{K+1}^T. \tag{2.25}$$

We can expand this as follows:

$$\begin{aligned}\mathbf{R}_{K+1} &= \begin{bmatrix} \mathbf{s}_K \\ s[n-1-K] \end{bmatrix} [\mathbf{s}_K^T \quad s[n-1-K]] \\ &= \begin{bmatrix} \mathbf{R}_K & \mathbf{r}_K \\ \mathbf{r}_K^T & \rho_K \end{bmatrix}\end{aligned} \tag{2.26}$$

where

$$\mathbf{r}_K = \mathbf{s}_K s[n-1-K] = \begin{bmatrix} r[K] \\ r[K-1] \\ \vdots \\ r[1] \end{bmatrix} \tag{2.27}$$

and

$$\rho_K = s^2[n-1-K]. \tag{2.28}$$

Here $\mathbf{r}_K$ is the cross-correlation between the past K samples and the $(K+1)^{\text{th}}$ past sample; ρ_K is the power of the $(K+1)^{\text{th}}$ past sample. As can be seen from Eq. (2.26), the correlation matrix of the past K samples is equal to the K-by-K matrix obtained by the first K rows and columns of the correlation matrix of the past $K+1$ samples. Similarly,

$$\mathbf{d}_{K+1} = \mathbf{s}_{K+1}s[n] \tag{2.29}$$

$$\begin{aligned} &= \begin{bmatrix} \mathbf{s}_K \\ s[n-1-K] \end{bmatrix} s[n] \\ &= \begin{bmatrix} r[1] \\ r[2] \\ \vdots \\ r[K+1] \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{d}_K \\ d_{K+1}[n] \end{bmatrix}. \end{aligned} \tag{2.30}$$

The cross-correlation vector between the current sample and the past K samples is equal to the first K rows of the cross-correlation vector between the current sample and the past $K+1$ samples.

Inversion of Partitioned Hermitian Matrices

If we know $\mathbf{R}_K^{-1}$, then we can calculate $\mathbf{R}_{K+1}^{-1}$ using this information and the total complexity is less then computing the inverse of $\mathbf{R}_{K+1}$ from scratch. Let's denote the inverse of $\mathbf{R}_{K+1}$ with $\mathbf{Q}_{K+1}$. Since $\mathbf{Q}_{K+1}$ is also Hermitian, it can be partitioned the same way as $\mathbf{R}_{K+1}$ [6].

$$\mathbf{Q}_{K+1} = \begin{bmatrix} \mathbf{Q}_K & \mathbf{q}_K \\ \mathbf{q}_K^T & q_K \end{bmatrix} \tag{2.31}$$

$$\mathbf{R}_{K+1}\mathbf{Q}_{K+1} = \begin{bmatrix} \mathbf{R}_K & \mathbf{r}_K \\ \mathbf{r}_K^T & \rho_K \end{bmatrix} \begin{bmatrix} \mathbf{Q}_K & \mathbf{q}_K \\ \mathbf{q}_K^T & q_K \end{bmatrix} = \begin{bmatrix} \mathbf{I}_K & \mathbf{0}_K \\ \mathbf{0}_K^T & 1 \end{bmatrix}. \tag{2.32}$$

To find $\mathbf{Q}_{K+1}$, we need to solve the following set of four equations

$$\mathbf{R}_K\mathbf{Q}_K + \mathbf{r}_K\mathbf{q}_K^T = \mathbf{I}_K \tag{2.33}$$

$$\mathbf{r}_K^T\mathbf{Q}_K + \rho_K\mathbf{q}_K^T = \mathbf{0}_K^T \tag{2.34}$$

$$\mathbf{R}_K\mathbf{q}_K + \mathbf{r}_K q_K = \mathbf{0}_K \tag{2.35}$$

$$\mathbf{r}_K^T\mathbf{q}_K + \rho_K q_K = 1. \tag{2.36}$$

Using Eq. (2.35) we obtain

$$\mathbf{q}_K = -\mathbf{R}_K^{-1}\mathbf{r}_K q_K; \tag{2.37}$$

then substituting this in Eq. (2.36) we obtain

$$q_K = \frac{1}{\rho_K - \mathbf{r}_K^T\mathbf{R}_K^{-1}\mathbf{r}_K}. \tag{2.38}$$

Substituting this into Eq. (2.37) we get

$$\mathbf{q}_K = \frac{-\mathbf{R}_K^{-1}\mathbf{r}_K}{\rho_K - \mathbf{r}_K^T\mathbf{R}_K^{-1}\mathbf{r}_K}. \tag{2.39}$$

We can modify Eq. (2.33) to obtain

$$\mathbf{Q}_K = \mathbf{R}_K^{-1} - \mathbf{R}_K^{-1}\mathbf{r}_K\mathbf{q}_K^T. \tag{2.40}$$

Using Eq. (2.39) in this equation we obtain

$$\mathbf{Q}_K = \mathbf{R}_K^{-1} + \frac{-\mathbf{R}_K^{-1}\mathbf{r}_K(-\mathbf{R}_K^{-1}\mathbf{r}_K)^T}{\rho_K - \mathbf{r}_K^T\mathbf{R}_K^{-1}\mathbf{r}_K}. \tag{2.41}$$

To simplify this equation, we need to define two terms [6]

$$\begin{aligned}\mathbf{b}_K &= [b_0^K \quad b_2^K \quad \dots \quad b_{K-1}^K]^T \\ &= -\mathbf{R}_K^{-1}\mathbf{r}_K\end{aligned} \tag{2.42}$$

and

$$\begin{aligned}\alpha_K &= \rho_K - \mathbf{r}_K^T\mathbf{R}_K^{-1}\mathbf{r}_K \\ &= \rho_K - \mathbf{r}_K^T\mathbf{b}_K.\end{aligned} \tag{2.43}$$

Using Eqs. (2.42) and (2.43) in Eqs. (2.37) and (2.38) we can obtain

$$\mathbf{q}_K = \frac{\mathbf{b}_K}{\alpha_K} \tag{2.44}$$

$$q_K = \frac{1}{\alpha_K}. \tag{2.45}$$

Using Eqs. (2.37), (2.38), (2.42) and (2.43), and rearranging Eq. (2.41) we obtain the final equation representing $\mathbf{R}_{K+1}^{-1}$

$$\begin{aligned}\mathbf{R}_{K+1}^{-1} &= \begin{bmatrix} \mathbf{R}_K & \mathbf{r}_K \\ \mathbf{r}_K^T & \rho_K \end{bmatrix}^{-1} \\ &= \begin{bmatrix} \mathbf{R}_K^{-1} & \mathbf{0}_K \\ \mathbf{0}_K^T & 0 \end{bmatrix} + \frac{1}{\alpha_K}\begin{bmatrix} \mathbf{b}_K \\ 1 \end{bmatrix}[\mathbf{b}_K^T \quad 1].\end{aligned} \tag{2.46}$$

As can be seen from Eq. (2.46), $\mathbf{R}_{K+1}^{-1}$ can be calculated using $\mathbf{R}_K^{-1}$, $\mathbf{r}_K$ and ρ_K. As defined before, $\mathbf{r}_K$ is the cross-correlation between the past K samples and the $(K+1)^{\text{th}}$ past

sample; ρ_K is the power of the $(K+1)^{\text{th}}$ past sample. In other words, we do not need to calculate the inverse of $\mathbf{R}_{K+1}$ from scratch since we can easily calculate it using information we already have. Hence, starting with the inverse of $\mathbf{R}_2$, we can easily obtain the inverse of $\mathbf{R}_{K+1}$ recursively, which is computationally quite efficient.

The minimum mean square error estimator was found in Eq. (2.24). Solving it for a $(K+1)^{\text{th}}$ order MMSE estimator we get

$$\mathbf{a}^o_{K+1} = \mathbf{R}^{-1}_{K+1}\mathbf{d}_{K+1}. \tag{2.47}$$

Using equations (2.30) and (2.46), we can rewrite this equation as follows:

$$\begin{aligned}
\mathbf{a}^o_{K+1} &= \begin{bmatrix} \mathbf{R}_K^{-1} & \mathbf{0}_K \\ \mathbf{0}_K^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{d}_K \\ d_{K+1}[n] \end{bmatrix} + \frac{1}{\alpha_K} \begin{bmatrix} \mathbf{b}_K \\ 1 \end{bmatrix} \begin{bmatrix} \mathbf{b}_K^T & 1 \end{bmatrix} \begin{bmatrix} \mathbf{d}_K \\ d_{K+1}[n] \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{R}_K^{-1}\mathbf{d}_K \\ 0 \end{bmatrix} + \begin{bmatrix} \mathbf{b}_K \\ 1 \end{bmatrix} \frac{\mathbf{b}_K^T\mathbf{d}_K + d_{K+1}[n]}{\alpha_K} \\
&= \begin{bmatrix} \mathbf{a}^o_K \\ 0 \end{bmatrix} + \begin{bmatrix} \mathbf{b}_K \\ 1 \end{bmatrix} \kappa_K,
\end{aligned} \tag{2.48}$$

where

$$\kappa_K = \frac{\beta_K}{\alpha_K} \tag{2.49}$$

and

$$\beta_K = \mathbf{b}_K^T\mathbf{d}_K + d_{K+1}[n]. \tag{2.50}$$

This clearly shows that the $(K+1)^{\text{th}}$ order MMSE estimator can easily be calculated using the K^{th} order MMSE estimator.

2.2.4 Linear Predictive Coding

Linear predictive coding (LP coding) is probably the most important representative of parametric coders and, as the name implies, it uses linear prediction. In linear predictive coding, there are three key points.

1. Find the LP coefficients which will model the synthesis filter (filter coefficients)

2. Determine whether the frame is voiced or unvoiced

3. If it is determined that the frame is voiced, estimate the pitch period

Figure 2.4 shows a block diagram of linear predictive coding. The LP parameters are used to design the synthesis filter. Then if it is determined that the frame is unvoiced, white noise is generated to model the excitation signal, otherwise a pitch estimation is carried out which is then used to create a periodic signal. Hence the only information that is required to reproduce the speech signal at the receiver is LP coefficients, voiced/unvoiced indicator and pitch period. Linear predictive coding uses a very simplified model for speech production. Hence, while it achieves very low bit-rates, the quality of speech is very poor.

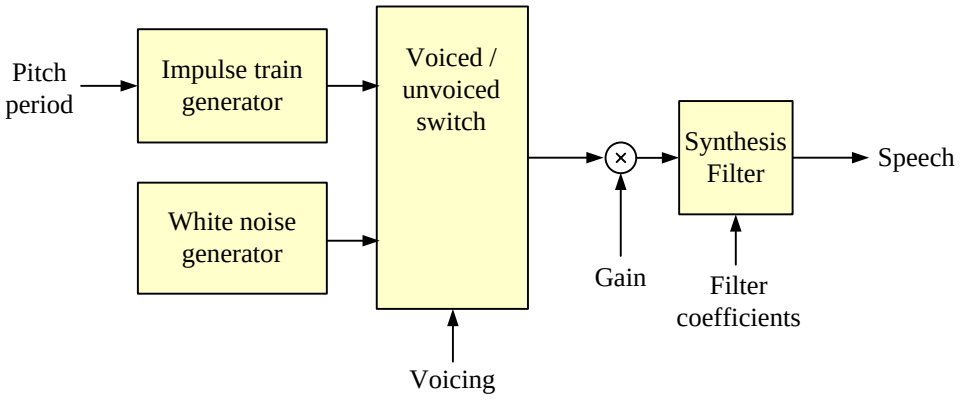


Fig. 2.4 The linear predictive coding model of speech production

2.3 Hybrid Coders

Hybrid coders combine the strengths of parametric and waveform coders. They use a speech production model like parametric coders to achieve low bit-rates and, similar to waveform coders, they attempt to match the decoded signal with the original signal in the time domain. The difference between parametric coders and hybrid coders is that parametric coders do not allocate any bits for the excitation signal — in the simplest version of LP coders, a periodic signal is used as the excitation signal if the speech frame is determined to be voiced and white noise is used if it is determined to be unvoiced. However, hybrid coders try to find the optimum excitation signal in order to match the decoded signal with the original waveform; hence they allocate bits for the excitation signal. In fact, most of the bits are allocated for the excitation signal in this type of coders. Thus, the bit-rate of

hybrid coders is between the bit-rates of parametric and waveform coders, but they achieve a better quality than both. ITU-T G.729 [7] and ITU-T G.723.1 [8] are two examples of hybrid coders. The most successful representatives of hybrid speech coders rely on code excited linear prediction (CELP)[9].

2.3.1 Code Excited Linear Prediction (CELP)

In the first versions of LP coders a prediction is made to determine if the speech segment is voiced or unvoiced. For the voiced case, a periodic signal is used as the excitation signal and for the unvoiced case, a white gaussian noise is used. However, this is a very poor estimation of the excitation signal and thus the quality is quite poor. In CELP, a codebook is used to determine the optimum excitation signal; hence the name code-excited is used to describe the method. The optimum excitation signal is found by trying to select the excitation signal among the ones in the codebook that generates synthetic speech as close to the original signal as possible. The measurement is made in the time domain using techniques like signal-to-noise ratio (SNR). A perceptually weighted error signal is used in this measurement. Therefore synthetically generated speech is tried to match the original waveform not only in the frequency domain (which is achieved by estimating the optimum LP parameters) but in the time domain, as well. The logic is simple. To reproduce a speech signal, we need to model the vocal tract and the excitation signal. Linear prediction is used to find the parameters that will model the speech synthesis filter (vocal tract) and a codebook is used to find the optimum excitation signal. Then LP coefficients and the index of optimum excitation signals used in the generation of speech signal are coded and sent. Using the same codebook at the receiver end, upon receiving the index and the LP parameters, the speech signal can be reproduced.

Analysis-by-synthesis

The method of finding the optimum excitation signal is called analysis-by-synthesis. In an open-loop system the parameters to be used in the reproduction of a speech signal are coded and sent. A more effective model is to use these parameters to generate the signal in the encoder and by comparing it with the original signal to find the best set of parameters that will match the synthetic speech as much as possible with the original signal. The procedure of finding the parameters to be used in synthesis is called analysis. Since the

signal is synthesized during the analysis to find the best parameters, this method is called analysis-by-synthesis.

The biggest problem of CELP is the high computational complexity of the codebook search process. There have been many ideas proposed to handle this problem. Algebraic code excited linear prediction (ACELP) [10] is one of the most famous ones.

2.3.2 Algebraic Code Excited Linear Prediction (ACELP)

The biggest problem of CELP, as discussed in the previous section, is the high complexity of the excitation codebook search. The memory requirement to store the excitation codebook, although not as crucial, is another drawback of CELP. There has been considerable amount of research to bring down the computational cost of CELP to practically applicable levels. Algebraic code excited linear prediction (or Algebraic CELP) is one of the most famous methods that researchers came up with to overcome these problems. Algebraic CELP, as the name implies, uses simple mathematical rules to create the excitation code-vectors. Since the code-vectors can easily be generated following simple mathematical rules, there is no need to actually store the codebook. The main logic is as follows: Each code-vector is composed of a predetermined number of pulses with predetermined and mutually exclusive possibility of positions and with the amplitude of either 1 or -1. Therefore, 1 bit per pulse is allocated for the amplitude. If we denote the number of possibilities of positions for the i^{th} pulse with N_i, and the number of pulses with K, then $\sum_{i=0}^{K-1} \log_2(N_i)$ bits are allocated to create the code-vectors.

2.3.3 Multi-Pulse Maximum Likelihood Quantization (MP-MLQ) Excitation

In MP-MLQ, a predetermined number of pulses is used for the excitation signal. To find the optimum position and amplitude of the pulses which will minimize the error, a maximum likelihood method is used.

2.4 ITU-T G.723.1: Dual Rate Speech Coder for Multimedia Communications

G.723.1 [8] is a dual rate speech coder designed for multimedia communications. It operates with a digital signal obtained by sampling an analog input at 8 kHz and then quantizing

the samples with 16-bit precision to obtain a linear PCM digital signal. The output of the decoder should be converted back to analog by similar means.

2.4.1 Modes of G.723.1

The G.723.1 speech coder can use 2 different methods to generate the excitation signal: algebraic code excited linear prediction (ACELP) and multi-pulse maximum likelihood quantization (MP-MLQ). The former gives a bit-rate of 5.3 kbit/s whereas the latter gives a bit-rate of 6.3 kbit/s. The user can choose one of these methods to code the whole speech signal as well as different methods for different frames.

Apart from these two methods, there is a voice activity detection and comfort noise generation (VAD/CNG) option which can be activated in addition to the choice of method for excitation signal generation. When this option is activated, whichever method is being used to generate the excitation signal, the coder recognizes a silence frame and uses fewer bits to represent those frames. The first packet is sent in the silence-insertion-description (SID) mode which updates the LP parameters and sets the comfort noise level for that first silence frame following an active frame. The succeeding packets corresponding to the succeeding silence frames are sent in the null mode. The decoder, upon receiving packets sent in SID and null modes, generates comfort noise using the parameters sent with the packet in the SID mode, which then replaces those silence frames. The bit-rate obviously decreases when the VAD/CNG mode is used, but this reduction is not taken into account in the overall bit-rate of the coder. To summarize, the coder generates packets in four different modes:

1. MP-MLQ: Multi-pulse coding at 6.3 kbit/s
2. ACELP: ACELP coding at 5.3 kbit/s
3. SID: Silence insertion description, updates the LP parameters and sets the comfort noise level
4. NULL: No data is sent in this mode

The coder is based on the principles of linear prediction analysis-by-synthesis coding, which is discussed in previous sections, and attempts to minimize a perceptually weighted error signal. The encoder operates on frames of 30 ms. Since the analog signal is sampled at

8 kHz, each frame of 30 ms has 240 samples. Each frame is first high pass filtered to remove the DC component. The high pass filter has the form

$$H_{HP}(z) = \frac{1 - z^{-1}}{1 - \frac{127}{128} z^{-1}}. \tag{2.51}$$

2.4.2 LP Analysis

The highpass filtered frame is then divided into four subframes of 60 samples each. For every subframe, a 10^{th} order linear predictive analysis is carried out. For this purpose, a hamming window of 180 samples is centred on the subframe of interest (60 samples back, 60 samples over the subframe and 60 samples ahead). The positions of the windows on the subframes are shown in figure 2.5.

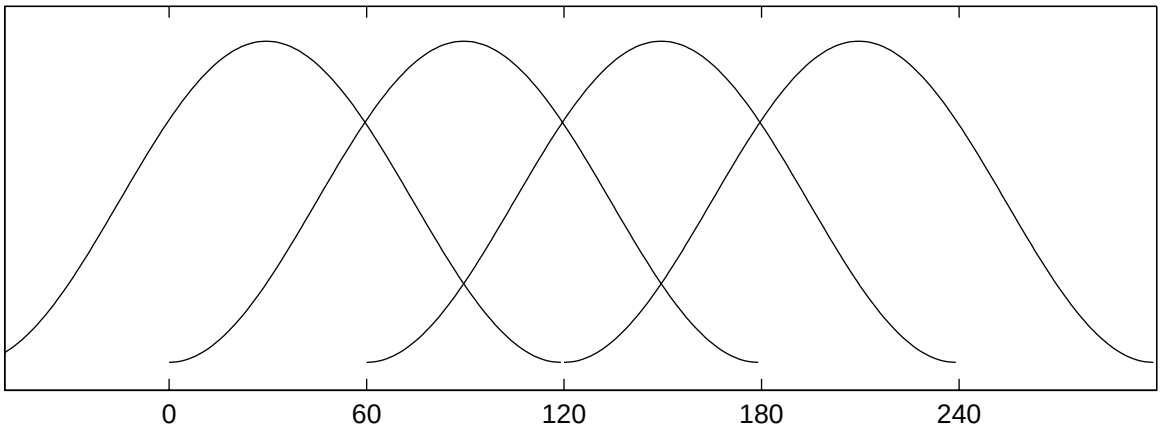


Fig. 2.5 LP windows

LP coefficients are computed using the Levinson-Durbin recursion. Since LP analysis is carried out on a subframe basis, 4 sets of LP coefficients, one for each subframe, are obtained for each frame. The fourth set of LP coefficients is converted to LSP (line spectral pairs) coefficients, which is represented with $\mathbf{p}'_n$:

$$\mathbf{p}'_n = \begin{bmatrix} p'_{1,n} \\ p'_{2,n} \\ \vdots \\ p'_{10,n} \end{bmatrix}. \tag{2.52}$$

The long term DC component is removed from the LSP coefficients and a new DC removed LSP vector $\mathbf{p}_n$ is obtained:

$$\mathbf{p}_n = \begin{bmatrix} p_{1,n} \\ p_{2,n} \\ \vdots \\ p_{10,n} \end{bmatrix}. \tag{2.53}$$

A first order fixed predictor, which has the value of 12/32 is applied to the previously decoded LSP vector $\tilde{\mathbf{p}}_{n-1}$ to obtain a DC removed predicted LSP vector $\bar{\mathbf{p}}_n$:

$$\bar{\mathbf{p}}_n = \begin{bmatrix} \bar{p}_{1,n} \\ \bar{p}_{2,n} \\ \vdots \\ \bar{p}_{10,n} \end{bmatrix} \tag{2.54}$$

$$\bar{\mathbf{p}}_n = \frac{12}{32}[\tilde{\mathbf{p}}_{n-1} - \mathbf{p}_{DC}]. \tag{2.55}$$

The difference between the DC removed LSP vector and the DC removed predicted LSP vector gives the LSP error vector:

$$\mathbf{e}_n = \mathbf{p}_n - \bar{\mathbf{p}}_n. \tag{2.56}$$

The Unquantized LSP vector $\mathbf{p}'_n$, the quantized LSP vector $\bar{\mathbf{p}}_n$ and the residual LSP error vector $\mathbf{e}_n$ are divided into 3 sub-vectors and vector quantized. Three indices are selected according to an error minimization criterion. This is called predictive split vector quantization. These 3 indices are transmitted. The transmitted indices correspond to the fourth set of LP coefficients. Hence, only the fourth set of LP coefficients of each frame is used in the construction of the synthesis filter. After the decoded LSP vector $\tilde{\mathbf{p}}_n$ is calculated, to obtain four sets of LP coefficients to be used for the synthesis of the speech corresponding to the current frame, linear interpolation is carried out using the decoded LSP vector $\tilde{\mathbf{p}}_n$ and the previous LSP vector $\tilde{\mathbf{p}}_{n-1}$. The 4 sets of unquantized LP coefficients are used to form a formant perceptual weighting filter, which is then used to weight the error signal during the search for the best excitation signal.

2.4.3 Generating Excitation Signal

Two pitch lag estimates are computed for every frame using the perceptually weighted speech. One estimate for the first two subframes, and one for the last two (it operates on a block of 120 samples). The open loop pitch estimate is not used directly in the decoding process (it is not transmitted), but is used to generate the harmonic noise weighting filter. The combination of synthesis filter, formant weighting filter and harmonic noise weighting filter forms the overall perceptual filter, and is used for closed loop analysis. The following figure shows a block diagram of the analysis-by-synthesis excitation signal search (Fig. 2.6).

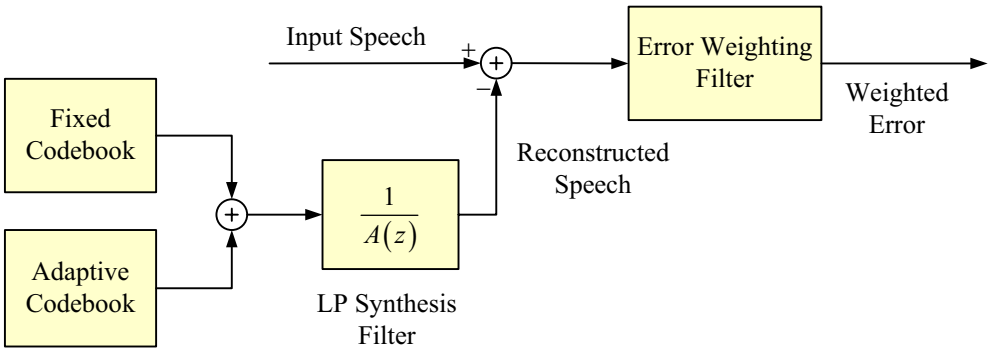


Fig. 2.6 Analysis-by-synthesis CELP coding

Adaptive Codebook

The adaptive codebook supplies the pitch contribution to the excitation signal. A fifth order pitch predictor is used. For subframes 0 and 2, the closed loop pitch lag is selected within a limited range (-1 to $+1$) around the open loop pitch lags found earlier. They are coded with 7 bits. For subframes 1 and 3, the closed loop pitch lags are calculated relative to the previous subframe and coded with 2 bits — they can differ from the previous lag only by -1, 0, $+1$ or $+2$. Adaptive codebook coefficients are taken from one of the two codebooks; the first has 85 entries, the second has 170. The first codebook is used for short pitch lags while the second one is used for longer pitch lags.

Fixed Codebook

A fixed codebook supplies the missing details in the excitation. G.723.1 is a dual rate speech coder with the options of 6.3 kbit/s and 5.3 kbit/s. The user can chose one of the two options for the fixed codebook contribution; the Multi-Pulse Maximum Likelihood Quantization (MP-MLQ) method to give the higher bit-rate of 6.3 kbit/s or the Adaptive Code Excited Linear Prediction (ACELP) method to give the lower bit-rate of 5.3 kbit/s.

High rate excitation (MP-MLQ) Multipulse excitation uses 6 pulses per subframe for subframes 0 and 2, and 5 pulses per subframe for subframes 1 and 3. The pulse positions are restricted — they can all be either on even-numbered positions or on odd-numbered positions. This is specified by a grid bit. Since a subframe has a length of 60 samples, and so does the excitation signal, there are 30 pulse positions (30 odd-numbered positions and 30 even-numbered positions) in which to place 5 or 6 pulses. The pulses for a subframe all have the same amplitude which is one of the 24 quantized values, but signs can change and are specified with 1 bit per pulse.

Low rate excitation (ACELP) Each fixed codevector, for this mode of operation, contains at most four non-zero pulses. The signs and positions that these pulses can assume are given in Table 2.1. There are 60 positions in total. The last possible positions of the last two pulses, which are given in parenthesis, refer to a non-present pulse. Hence a pulse at this position means that the pulse does not exist, which is how each fixed codevector containing 2-to-4 pulses is realized. The positions of all pulses can simultaneously be shifted by one to assume odd positions, which is realized by the use of an extra bit.

Table 2.1 ACELP excitation codebook

Sign	Positions
±1	0, 8, 16, 24, 32, 40, 48, 56
±1	2, 10, 18, 26, 34, 42, 50, 58
±1	4, 12, 20, 28, 36, 44, 52, (60)
±1	6, 14, 22, 30, 38, 46, 54, (62)

2.4.4 Bit allocation for the G.723.1 Speech Coder

G.723.1 has 4 modes of operations. Two of these modes are associated with the choice of method used for the fixed codebook contribution. Hence, for these two modes, the number of bits allocated for LPC indices, gains and adaptive codebook lags are the same. The remaining two modes are associated with the VAD/CNG option. When this option is activated, if a silence frame occurs, the SID mode is activated automatically in which LPC parameters and the SID gain are sent. For the succeeding silence frames, no data is sent. For all of these modes, 2 bits are allocated to indicate the mode of operation of the coder.

High rate excitation (MP-MLQ)

There are 30 possible positions per 6 pulses for subframes 0 and 2 and there are 30 possible positions per 5 pulses for subframes 1 and 3. Therefore $\log_2\binom{30}{6} = 19.18 \approx 20$ bits are allocated for each of the subframes 0 and 2, whereas $\log_2\binom{30}{5} = 17.12 \approx 18$ bits are allocated for each of the subframes 1 and 3. This gives a total of 76 bits. By using the fact that the number of codewords in the fixed codebook is not a power of 2, 3 additional bits are saved by combining the 4 MSB (Most Significant Bits) of each pulse position index into a single 13-bit word [8]. Hence the total number of bits allocated for pulse positions for this mode of operation is 73 bits. One bit per pulse is used to indicate the sign and 1 bit per subframe is used as the grid information to indicate whether the pulses will assume the odd-numbered positions or the even-numbered positions. The total number of bits allocated per frame is 189 in this mode. For a frame of 30 msec, this results in a bit-rate of 6.3 kbit/s. The details of the bit allocation for the Multi-Pulse Maximum Likelihood Quantization method is given in Table 2.2.

Table 2.2 Bit allocation of the 6.3 kbit/s coding algorithm

Parameters Coded	Subframe 0	Subframe 1	Subframe 2	Subframe 3	Total
LPC indices					24
Adaptive codebook lags	7	2	7	2	18
All the gains combined	12	12	12	12	48
Pulse positions	20	18	20	18	73
Pulse signs	6	5	6	5	22
Grid index	1	1	1	1	4
Total					189

Low rate excitation (ACELP)

As can be seen in Table 2.1, there are 8 possible positions per pulse, which requires 3 bits, and 1 bit is required for the sign. Hence, 12 bits per subframe are allocated for pulse positions and 4 bits per subframe are allocated for the signs. 1 extra bit per subframe is allocated for the grid information, which is used to indicate whether the pulses assume even or odd positions. The total number of bits allocated per frame is 158 for this mode. This results in a bit-rate of 5.3 kbit/s for a frame of 30 msec. The bit allocation for low rate excitation can be seen in Table 2.3.

Table 2.3 Bit allocation of the 5.3 kbit/s coding algorithm

Parameters Coded	Subframe 0	Subframe 1	Subframe 2	Subframe 3	Total
LPC indices					24
Adaptive codebook lags	7	2	7	2	18
All the gains combined	12	12	12	12	48
Pulse positions	12	12	12	12	48
Pulse signs	4	4	4	4	16
Grid index	1	1	1	1	4
Total					158

Silence Frame

When the VAD/CNG option is activated, if a silent frame occurs, 24 bits corresponding to the LPC coefficients and 6 bits for the SID (silence insertion descriptor) gain are sent. This makes a total of 30 bits. There is an additional 2 bits for the SID mode indicator. The succeeding silence frames are considered as null and only the 2 mode-indicating-bits are sent for null frames.

Table 2.4 summarizes the bit allocation for packets generated at different modes.

2.5 Perceptual Evaluation of Speech Quality (PESQ)

To evaluate the performance of G.723.1, Perceptual Evaluation of Speech Quality (PESQ) is used. PESQ is described in the ITU standard P.862. PESQ compares the quality of 2 input speech signals:

Table 2.4 Summary of the bit allocation for packets generated at different modes

Frame Type	Number of Data Bits	Mode Indicating Bits	Total Number of Bits
MP-MLQ	190[1]	2	192
ACELP	158	2	160
SID	30	2	32
NULL	0	2	2

1. original signal
2. degraded signal that is obtained by passing the original signal through a communications system

The output of PESQ is the prediction of the perceived quality of the degraded signal in terms of a score on a scale 1 to 5 that would be given by subjects in a subjective listening test by comparing the degraded signal with the original one. For more details, the reader is encouraged to refer to [11].

2.6 Performance of G.723.1 Measured in PESQ

To test the performance of G.723.1 in terms of PESQ scores, 22 speech files, consisting of 11 different sentence groups recorded by 11 different female and 11 different male speakers are used. These files are obtained by concatenating 4 utterances of each speaker. Each concatenated speech file is approximately 10 sec long. Each speech file is coded with four different modes:

- 5.3 kbit/s VAD/CNG disabled
- 5.3 kbit/s VAD/CNG enabled
- 6.3 kbit/s VAD/CNG disabled
- 6.3 kbit/s VAD/CNG enabled

The results are illustrated in Fig. 2.7. The first 11 indices of the x-axis in Fig. 2.7 refer

[1]There is one unused bit in the MP-MLQ mode, which is referred to as the reserved bit.

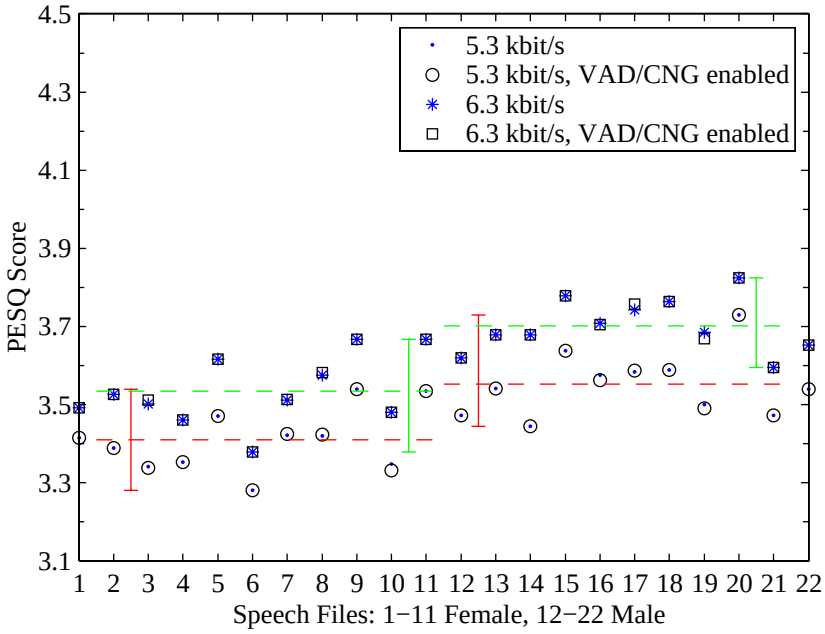


Fig. 2.7 PESQ results for 11 female and 11 male speakers and different modes

to 11 female speakers and the last 11 indices refer to the male speakers. As can be seen, the average, maximum and minimum of the PESQ scores for males are higher than those for females. The exact results (average, maximum and minimum) for different modes are given in Table 2.5. In addition to testing the decoded speech quality, a test is made to find

Table 2.5 PESQ scores for different modes of G.723.1

Speech Signal	Coder Mode	VAD/CNG	Min PESQ	Avg PESQ	Max PESQ
Male	5.3 kbit/s	disabled	3.44	3.55	3.73
Male	5.3 kbit/s	enabled	3.44	3.55	3.73
Male	6.3 kbit/s	disabled	3.60	3.70	3.83
Male	6.3 kbit/s	enabled	3.60	3.70	3.83
Female	5.3 kbit/s	disabled	3.28	3.41	3.54
Female	5.3 kbit/s	enabled	3.28	3.41	3.54
Female	6.3 kbit/s	disabled	3.38	3.53	3.67
Female	6.3 kbit/s	enabled	3.38	3.54	3.67

the highest possible PESQ score. When the reference file is used both as the clean signal and as the degraded signal, a PESQ score of 4.5 is obtained. Hence, the PESQ scores given

in Table 2.5 and Fig. 2.7 are out of 4.5. These results can be interpreted as follows:

- The G.723.1 speech coder has a better performance at 6.3 kbit/s than at 5.3 kbit/s.
- The G.723.1 speech coder has a better performance for male speech than for female speech.
- Enabling VAD/CNG (voice activity detection and comfort noise generation) mode does not reduce the quality.

The performance of a coder in general on male and female speech is coder dependent. Some coders are better at coding female speech than male speech and some coders are better at coding male speech. The observation that G.723.1 gives a better performance at 6.3 kbit/s makes sense. The two modes only differ at the fixed codebook contribution for the excitation signal. Although the algorithms are different, more bits are allocated for the MP-MLQ mode than the ACELP mode, which creates the tradeoff between bit-rate and quality. It does make sense that the quality does not decrease despite the decrease in the bit-rate upon activating the VAD/CNG mode. Upon enabling the VAD/CNG option, silence frames are detected and instead of coding those frames with whichever bit-rate is selected for the coding operation, LPC parameters and a gain parameter are sent which are used to create comfort noise to replace the silent frames. This procedure saves bit-rate and since silence frames do not include any speech content, not applying the selected coding method does not decrease the quality.

2.7 Chapter Summary

In this chapter, we first talked about waveform, parametric and hybrid speech coders. Waveform coders are the simplest speech coders. They try to preserve the waveform by directly coding the speech samples. Parametric coders, on the other hand, try to model the speech synthesis. Many parametric speech coders use linear prediction (LP) to model the vocal tract. Speech signals are referred to as quasi-stationary since their characteristics change in time but remain relatively unchanged for a short period of time. Therefore parametric coders operate on a frame basis. They find the parameters to model the vocal tract for each frame and depending on whether the frame is voiced or unvoiced, they use either a periodic signal or white noise to model the excitation signal. For each frame,

in addition to the LP parameters, a voiced / unvoiced indicator is sent along with the pitch estimate if the frame is determined to be voiced. Hybrid coders are a combination of waveform and parametric coders — they attempt to find the parameters to model the synthesis of each frame of a speech signal while also providing an excitation signal that minimizes the error in some sense to drive this model. Hybrid coders combine the strengths of waveform and parametric coders, therefore many modern coders are hybrid.

We then talked about the G.723.1 speech coder, since we used it as the test platform in this research. G.723.1 is a dual rate hybrid speech coder designed for multimedia communication. It operates on frames of 30 ms. G.723.1 can use two different methods to generate excitation signal; algebraic code excited linear prediction (ACELP) and multi-pulse maximum likelihood quantization (MP-MLQ). The former gives a bit-rate of 5.3 kbit/s (158 bits per frame: 24 bits for LP parameters, 134 bits for excitation parameters) whereas the latter gives a bit-rate of 6.3 kbit/s (189 bits per frame: 24 bits for LP parameters, 165 bits for excitation parameters).

At the end of the chapter we illustrated the performance of the G.723.1 speech coder in different modes, in terms of PESQ scores. We observed that its performance changes depending on the gender of the speaker and the mode of the coder.

Chapter 3

Packet-Loss-Concealment Schemes

In digital communication, data is converted to bits by using coders and in most cases it is sent as a bitstream, so the bit order is preserved. Speech coders aim to represent a speech signal with very few bits while maintaining a toll quality. They can achieve very low bit-rates by taking advantage of the redundancy in speech signals — they use past information to encode and decode current information. However, speech coding algorithms are not robust to transmission errors [12]. The Internet is a packet-switched, best effort delivery service, in which the quality of service is not guaranteed [12]. After speech is coded, the bitstream is divided into packets and sent in packets. Packets experience variable delays, which necessitates the use of a buffer. A receiver buffer holds a packet until a scheduled playout time. A packet is considered lost if it does not arrive before its scheduled playout time. When a packet loss occurs, due to the dependence of the decoding of a frame on previous frames, error propagates to subsequent frames [13]. Therefore modern speech coders have packet-loss-concealment schemes to deal with the problem of packet loss.

3.1 Receiver-Based Schemes

Receiver-based schemes perform loss concealment procedures independent of the sender. The simplest methods replace the segments of the speech that the lost packets correspond to with silence or white noise [14] or repeat the last received packet [15]. These methods do not really aim to regenerate the lost waveform. They rely on the assumption that the losses are not frequent or consecutive. However, this is a naive assumption, which most of the time does not hold for the Internet [12]. These algorithms also rely on the assumption

that the effect of one lost packet does not propagate to succeeding packets, which does not hold for many modern coders. Hence, simple receiver-based schemes can only be applied if decoding of a frame does not depend on the successful transmission and decoding of previous frames. This condition holds for waveform coders. However, these schemes do not even work well for waveform coders, especially with frequent packet losses [16].

There are some more-sophisticated receiver-based schemes such as the one used in G.723.1. The packet-loss-concealment scheme of G.723.1 will be discussed later in this chapter.

3.2 Sender-Receiver-Based Schemes

Sender-receiver-based schemes are usually more effective than receiver-based schemes [12]. Senders can transmit knowledge about lost packets to receivers; hence receivers can make a better estimate of the lost packet and can actually attempt to regenerate the waveform that the lost data corresponds to. There are several methods in this category.

3.2.1 Priority-Based Schemes

There are some priority-based schemes which assign different priorities to different packets, thereby decreasing the probability of the high-prioritized packets being dropped. They set priorities according to signal energy, difference from previous packets and voiced/unvoiced indicators [17]; or according to whether or not a packet can be well reconstructed using previous packets [18], [19]. Priority-based schemes, clearly, require a supporting network — a network which drops packets according to the priorities assigned to them [12]. However, the Internet is not a priority-based network.

3.2.2 Redundancy-Based Schemes

Most of the promising sender-receiver-based methods rely on adding redundancy. Some information about each packet, according to the method being used, is sent either with the previous or with the next packet, which is then used to regenerate the lost waveform. The most naive way is to add copies of the previous K frames to frame n, so packet n has $K + 1$ frames [20]. For this method, $K = 1$, for instance, corresponds to duplicating each packet, which of course either increases the bandwidth by doubling the bit-rate or increases

the delay by doubling the packet size. Another similar method, which is called Forward Error Correction (FEC), sends an extra packet for each set of K packets that includes information which can be used to reproduce the data in any one of those K packets [21], [22]. There are also some redundancy-based schemes, which aim to protect part of the data. An example of possible redundant information is pitch information and a voiced/unvoiced indicator [23], [24]. If a packet is lost, a basic linear predictive coding can be applied to retrieve the lost information upon receiving the pitch and voiced/unvoiced indicator — as discussed in previous sections, the only information needed to generate the excitation signal for linear predictive coding is the pitch information and the voiced/unvoiced indicator. The LP parameters of the last received packet can be used when a loss occurs.

3.2.3 Interleaving-Based Schemes

Interleaving-based schemes, as opposed to redundancy-based schemes, do not rely on adding redundancy. The idea is that consecutive speech samples are correlated with each other and by distributing the data in a given frame to several frames, the loss of a packet can be converted to random losses in several frames. When this happens, the only thing to do to recover the missing data is to apply interpolation. The disadvantage is that the loss of one packet spreads to several frames. For this idea to be applicable, the consecutive samples of the data to be sent through Internet must be correlated. Speech signals meet this condition, hence the method is applicable to waveform coders. However, for parametric coders and hybrid coders, it is not the samples of the waveform itself that are coded and sent, but some parameters which are used to reproduce a speech signal following a model. So for this method to be applicable to parametric and hybrid coders, LP parameters and excitation parameters must be correlated. Research shows that LP parameters are correlated with each other, but excitation parameters are not [25]. Hence, many successful interleaving methods applied to parametric and hybrid coders interleave the LP parameters and send the excitation parameters as redundant information [25]. However, especially in hybrid coders, most of the bits are allocated for excitation parameters. In addition to this, as will be shown in later sections, LP parameters are not that crucial in the packet-loss-concealment process — LP parameters of the previous frames can be used for the current frame. Because of these reasons, it is pointless to interleave LP parameters in addition to sending the excitation parameters twice, since just duplicating the excitation parameters

will do just as good, furthermore, additional delay will be avoided. Hence, interleaving is not a very good method for modern speech coders.

3.3 The Tolerance of Speech Coders to Packet Losses

Speech coders can tolerate up to 5% random losses when using packet-loss-concealment schemes [26]. However, even a single packet loss at a "critical" frame can be quite audible as we will see later.

3.4 Requirements of a Good Packet-Loss-Concealment Scheme

As discussed in previous sections, parametric coders take advantage of redundancy found in speech signals to reach very low bit-rates. As a result of this, for many parametric coders, the encoding and decoding of each frame depends on the information of the previous frames. The dependence on past frames to decode the current frame introduces the concept of coder state. After the decoding of each packet, some information is saved to be used in the decoding process of the next packet, which can also be referred to as state updating. This information usually includes past excitation parameters and LP coefficients. In other words, a decoder needs two sources of information to complete its task — information in the current frame and the state information. Therefore, when a packet is lost, decoding of that packet is affected due to the loss of the data in that packet. Moreover, since the states cannot be updated properly, the decoding of succeeding packets are affected, too, even if they are received properly. Therefore, there are two key features that a good packet-loss-concealment scheme to be used for parametric coders should have:

- It should be able to reconstruct a reasonable facsimile of the segment of the speech that the data in a lost packet corresponds to.
- It should be able to update the states of the subsequent packet well enough so as to mitigate the effect of the lost packet on succeeding frames.

3.5 Using Late Frames to Improve Packet Recovery

As discussed before, in Voice Over IP, data is sent in packets, and since the quality of service is not guaranteed in the Internet, packets may not arrive in order or may arrive too late for playout [12]. When a packet does not arrive in time for playout, it is considered lost. For a while, common practice has been to discard the late packets; however, research shows that these late packets are not useless [27]. On the contrary, they can be used to significantly improve the quality of the concealment. In [27], it is shown that late frames, when used to update the states, can be quite useful in the packet-loss-concealment process. Figure 3.1 is an illustration of their algorithm for one late frame. Line A shows the output without any

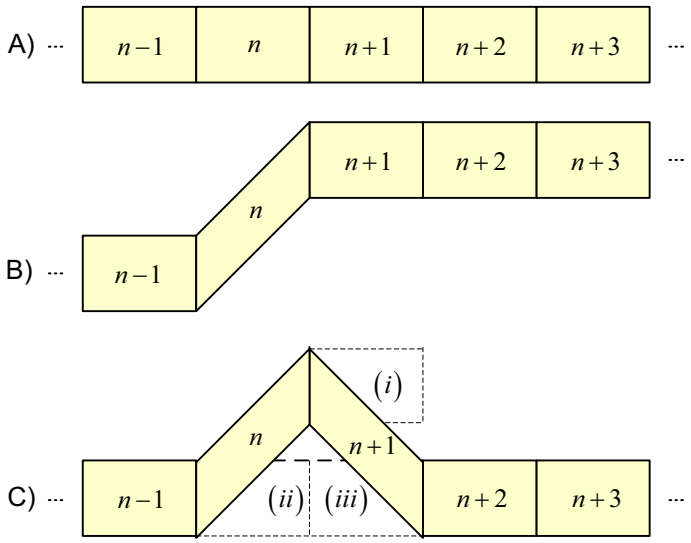


Fig. 3.1 Chronogram showing the effects of one late frame [27]

loss or late frame. Line B shows the effect of one late frame. Here, frame $n-1$ is received without any error, but frame n is late for playout. Hence a packet-loss-concealment scheme is applied to recover the information in frame n. However, the states cannot be updated perfectly. Packet $n+1$ is also received in time; however, due to the fact that the states could not be updated perfectly during the packet-loss-concealment procedure, wrong state information is used both in the decoding of frame $n+1$ and in the updating of the states of frame $n+2$. Hence, the process continues in a different state. Line C shows how the late packet can be used. Supposing that frame n arrives before playout of frame $n+1$, states of frame $n+1$ can be updated. To do this, upon receiving frame n, it is decoded only to

update the states of frame $n+1$, then frame $n+1$ is decoded as if no error has occurred, thereby bringing the state of the decoding process to where it should be. Of course this introduces some complexities. If we want to be able to go back one frame in time to update the states, we must allocate enough memory to store the states of one frame. In general, we must allocate enough memory to store the state information of as many frames as we want to go back. Additional required processing power changes depending on when the late frame arrives. Referring to Fig. 3.1 again, upon arrival of frame n, decoding of it should be repeated (referred to as (ii) in Fig. 3.1). In addition to this, if frame n arrives at the end of decoding of frame $n+1$, then the decoding of frame $n+1$ must also be repeated (referred to as (i) in Fig. 3.1). This research shows that using late frames in updating the states eliminates error propagation.

3.6 Packet-Loss-Concealment Scheme Used in G.723.1

In G.723.1, a counter is used to track the number of the lost packets and trigger the packet loss concealment. This counter is initiated as zero at the beginning of the decoding process. If a packet is lost, the counter is incremented by one and upon receiving a packet it is reset to zero. Two different schemes are used according to the number of successive losses. If the counter shows less than or equal to 3, one scheme is applied and if the counter shows greater than 3 another scheme is applied. The packet-loss-concealment scheme works in two steps:

1. Concealment of LP coefficients
2. Concealment of excitation parameters

3.6.1 Recovery of LP Coefficients

In G.723.1, as discussed in Chapter 2, LP coefficients are converted to LSPs (line spectral pairs), and it is the LSPs that are coded and sent. In the decoding of LSPs, using the 3 indices that are transmitted, the decoded residual LSP error vector $\tilde{\mathbf{e}}_n$ is obtained. Then the predicted vector $\bar{\mathbf{p}}_n$ is added to the decoded LSP error vector $\tilde{\mathbf{e}}_n$ and DC vector $\mathbf{p}_{DC}$ to form the decoded LSP vector $\tilde{\mathbf{p}}_n$:

$$\bar{\mathbf{p}}_n = \frac{12}{32}[\tilde{\mathbf{p}}_{n-1} - \mathbf{p}_{DC}] \tag{3.1}$$

$$\tilde{\mathbf{p}}_n = \bar{\mathbf{p}}_n + \tilde{\mathbf{e}}_n + \mathbf{p}_{DC}. \tag{3.2}$$

As can be seen from Eqs. (3.1) and (3.2), the previous decoded LSP is used in the computation of current decoded LSP. If the counter does not show 0, then 23/32 is used as the fixed predictor in the computation of $\bar{\mathbf{p}}_n$. This can be interpreted as increasing the effect of the previous decoded LSP on the computation of current decoded LSP, in case of a loss. Regardless of whether or not the packet of interest was lost, a stability check is performed on the decoded LSP vector $\tilde{\mathbf{p}}_n$ according to the following predefined condition.

$$\tilde{p}_{j+1,n} - \tilde{p}_{j,n} \geq \triangle_{\min}, \quad 1 \leq j \leq 9. \tag{3.3}$$

This can be interpreted as the difference between consecutive coefficients being less than a predetermined value. For the predetermined value $\triangle_{\min}$, 31.25 Hz is used if there is no loss, and 62.5 Hz is used if the counter does not show 0. The 10 LSP coefficients are modified according to the following scheme if they do not meet this condition:

$$\tilde{p}_{avg} = \left(\tilde{p}_j + \tilde{p}_{j+1}\right)/2 \tag{3.4}$$

$$\tilde{p}_j = \tilde{p}_{\text{avg}} - \triangle_{\min}/2 \tag{3.5}$$

$$\tilde{p}_{j+1} = \tilde{p}_{\text{avg}} + \triangle_{\min}/2. \tag{3.6}$$

The stability check is repeated 10 times until the condition is met. If the stability condition is not met after 10 iterations, the previous LSP vector is used.

3.6.2 Recovery of Excitation Parameters

G.723.1 allocates memory for past excitation parameters, the size of which is the predefined maximum pitch lag. If the counter shows a number greater than 3, the memory allocated for past excitation parameters and current excitation parameters is cleared. If the maximum allowed number of subsequent losses has not occurred, then one of the two methods is applied according to the frame type (voiced / unvoiced). The decision as to whether the frame is voiced or unvoiced is made according to the last previous good frame. If it is an unvoiced frame (a frame generated in either SID mode or NULL mode), then the lost frame is considered as a null frame and comfort noise is generated using the parameters received with the last good frame (silence frame generated in SID mode). If it is a voiced

frame then a periodic excitation signal is generated using the period that was previously found. The classifying of whether the frame is voiced or not is made based on a cross-correlation maximization method. The last 120 excitation parameters of the frame are cross-correlated around the previous pitch lag within a range of ±3. The lag which reaches the maximum correlation value is selected as the interpolation index candidate. If the previous pitch lag is L, then the pitch value that satisfies maximum correlation is in the range $[L-3, L+3]$. Then the prediction gain of the best vector is tested. If the gain is greater than 0.58 dB, the frame is declared as voiced, otherwise it is declared as unvoiced. If the frame is declared as unvoiced, each excitation parameter is generated by using a randomly generated number and a gain that was calculated previously. This procedure is called comfort noise generation. If, when this gain is calculated, the counter shows zero (indicates that the frame is received correctly), then gain is assigned using a table. If the counter does not show zero (indicates that the frame is lost), then the previous gain is attenuated by 2.5 dB to be used in the regeneration of the excitation signal ($10^{-2.5/20} = 0.75$: the previous excitation vector is multiplied by 0.75), if the frame is declared to be unvoiced. This attenuation can be repeated three times, since only three successive losses are allowed, after which the excitation memory is cleared (set to zero). If the frame is declared to be voiced, then the pitch lag determined previously is used to create a periodic signal. The last n excitation parameters are repeated to generate the excitation signal and this n is equal to the pitch lag.

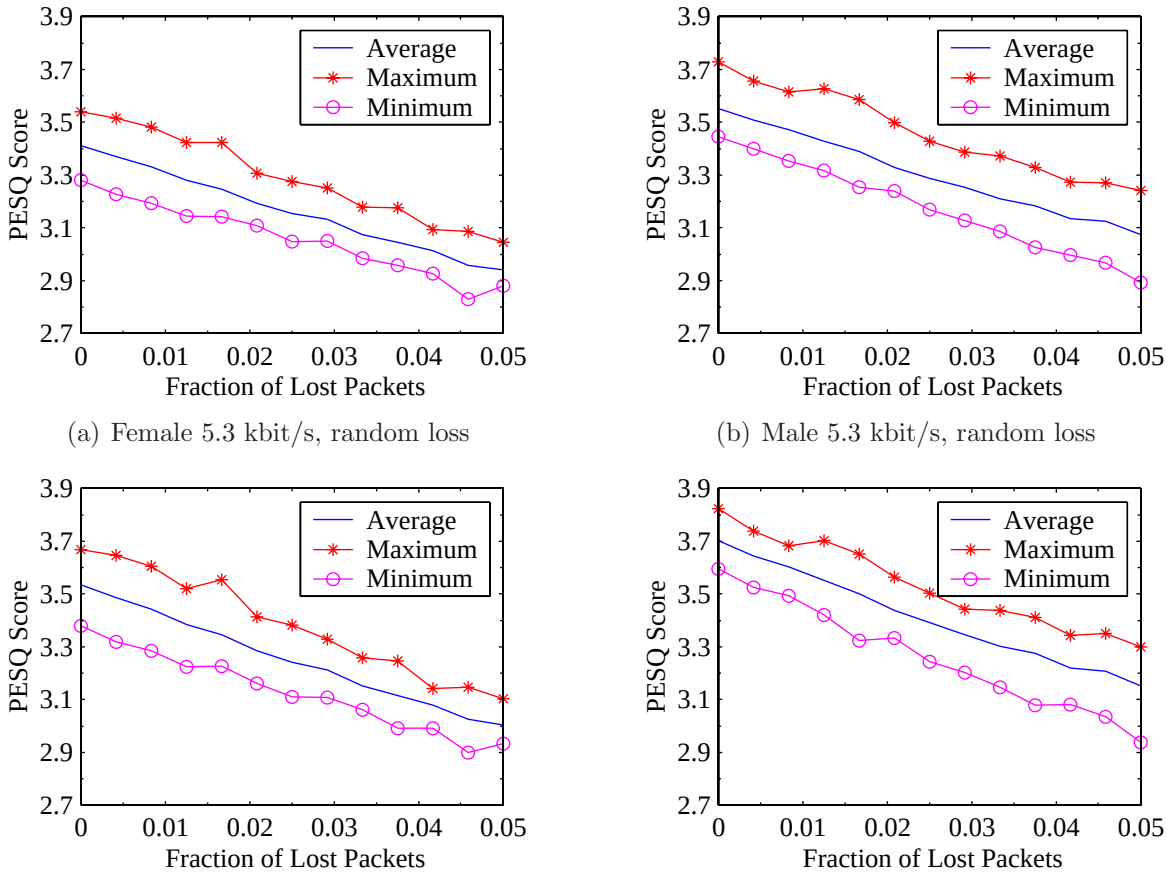
3.6.3 Performance of the Packet-Loss-Concealment Scheme of G.723.1 Measured in PESQ

The same 11 female and 11 male speech files are used to measure the performance of the packet-loss-concealment scheme of G.723.1. All the files are coded at the two possible bit-rates (5.3 kbit/s and 6.3 kbit/s). The VAD/CNG (voice activity detection and comfort noise generation) mode is disabled. Since each speech file is obtained by concatenating 4 utterances of the same speaker, each speech signal is active almost 100% of the time and therefore it does not make any change in our case to enable this mode.

For each file and each mode, PESQ scores are found for different loss scenarios (different number of lost packets). Following algorithm is used to determine the scenarios to be used for each file.

1. Let the number of frames in the shortest file be $N_{\min}$ The number of losses to be used for each scenario for the shortest speech file is determined as $0, 1, \ldots, L$ where $L = \lfloor 0.05 \times N_{\min} \rfloor$.
2. The percentages corresponding to these scenarios are calculated as $0 \times \frac{1}{N_{\min}}, 1 \times \frac{1}{N_{\min}}, \ldots, L \times \frac{1}{N_{\min}}$.
3. The number of lost packets for each scenario corresponding to these percentages are calculated for each speech file as $0 \times \left\lfloor \frac{N_k}{N_{\min}} \right\rfloor, 1 \times \left\lfloor \frac{N_k}{N_{\min}} \right\rfloor, \ldots, L \times \left\lfloor \frac{N_k}{N_{\min}} \right\rfloor$ where N_k is the number of frames of the k^{th} speech file.

This algorithm guarantees that the number of the lost packets in consecutive scenarios will be as close as possible but not the same. The first two scenarios correspond to no loss and 1 packet loss for every speech file. To find a PESQ score for a scenario (for a specified number of losses), the average of 10 PESQ scores found for 10 different cases is taken. The locations of the lost packets for these cases are determined randomly; however, consecutive losses are avoided. Since it was observed that the coder had a different performance for male and female speech, separate statistics are kept for male and female speakers. Hence the results are illustrated both for males and females. For a specified fraction of losses, an average PESQ score was found by taking the average of the PESQ scores of the speech files recorded by the same gender and coded at the same rate. The results are shown in Fig. 3.2. For a given loss scenario, the 3 curves correspond to the maximum PESQ score of the 11 speech files, the average of the PESQ scores and the minimum PESQ score. As expected, PESQ scores decrease as the number of losses increases. The rate of the decrease is not significantly different for any of the cases. As expected, the PESQ scores are better for male speech files than for female speech files and are better at 6.3 kbit/s than at 5.3 kbit/s. On average, at 5.3 kbit/s, PESQ scores decrease by 14% for female speech (3.41 for no loss and 2.94 for 5% loss) and 13% for male speech (3.55 for no loss and 3.08 for 5% loss). At 6.3 kbit/s, they decrease by 15% for female speech (3.53 for no loss and 3.00 for 5% loss) and 15% for male speech (3.70 for no loss and 3.15 for 5% loss). Table 3.1 summarizes these results.



(a) Female 5.3 kbit/s, random loss

(b) Male 5.3 kbit/s, random loss

(c) Female 6.3 kbit/s, random loss

(d) Male 6.3 kbit/s, random loss

Fig. 3.2 Performance of the Packet-Loss-Concealment Scheme of G.723.1

Table 3.1 PESQ scores for no loss and under 5% random loss

Speech Signal	Coder Mode	No Loss	5% Random Loss	Change
Female	5.3 kbit/s	3.41	2.94	14%
Male	5.3 kbit/s	3.55	3.08	13%
Female	6.3 kbit/s	3.53	3.00	15%
Male	6.3 kbit/s	3.70	3.07	15%

3.7 Dynamically Updating the Coder States

Many packet-loss-concealment schemes that rely on adding redundancy send redundant information regardless of how important each packet is. However, if the data in the lost packet can be recovered and is not crucial in updating the states, then we do not need to send extra information about that packet. In other words, the decision of whether or not to send extra information about each packet should be made depending on how important each packet is. This brings two questions to mind:

1. How can we determine how important each packet is?

2. What is the cost of implementing this algorithm?

The answers to these questions are the main focus of this research and will be given in Chapter 4.

3.8 Chapter Summary

In this chapter, we first talked about the packet loss problem. Speech coders can achieve very low bit-rates by taking advantage of the redundancy in speech signals — they use past information to encode and decode current information. However, speech coding algorithms are not inherently robust to transmission errors. For voice transmission over the Internet, after speech is coded, the bitstream is divided into packets and sent in packets. Packets experience variable network delays. Real-time voice transmission over the Internet necessitates a limit on the waiting time for the arrival of a packet. A receiver buffer is used to hold packets until their scheduled playout times — the packets which arrive after are considered lost. The dependence on past frames to decode the current frame introduces the concept of coder state. After the decoding of each packet, some information is saved (state update) to be used in the decoding process of the next packet. This information usually includes past excitation parameters and LP coefficients. When a packet loss occurs, due to the dependance of the decoding of a frame to previous frames, the error propagates to subsequent frames.

We talked about packet-loss-concealment schemes, which can be categorized in two groups: receiver-based schemes and sender-receiver-based schemes. Receiver-based schemes

try to reproduce the speech segment that a lost packet corresponds to by using the previous and subsequent segments of the speech or replace it with another waveform. Sender-receiver-based schemes are those which use the transmitter as well as the receiver for packet loss concealment. Sender-receiver-based schemes can further be categorized in three groups: priority-based schemes, redundancy-based schemes and interleaving-based schemes. Priority-based schemes assign priority to the packets according to their importance and assume that the packets will be dropped by a supporting network according to the preassigned priorities. Redundancy-based schemes add redundant information at the transmitter about each packet to either the previous or the next packet, which is then used in the receiver in case of a loss. Interleaving-based schemes distribute the information in a packet into several packets, so that when a packet is lost, only part of the information in that packet is gone and the lost information can be recovered using the part of the information that was distributed to other packets.

We talked about the two key features that a good packet-loss-concealment scheme to be used for parametric coders should have. It should be able to reconstruct a reasonable facsimile of the segment of the speech that the data in the lost packet corresponds to and it should be able to update the states of the subsequent packet so as to mitigate the effect of the lost packet on succeeding frames.

We then explained the details of the packet-loss-concealment scheme of G.723.1. At the end of the chapter we illustrated its performance in terms of PESQ scores. We observed that for 5% random losses, PESQ score drops by on average 15% from the no loss case.

Chapter 4

Experimental Results

Powerful packet-loss-concealment schemes have been proposed which rely on sending redundant information. They determine the extra information that is required to adequately regenerate the waveform that the data of each packet corresponds to in case they are lost. They send that extra information with either the previous or the next packet. However, not all the packets have the same importance. If the data in the lost packet is not crucial in updating the states and if the speech segment that it corresponds to can be adequately regenerated, then we do not need to send extra information about that packet. In other words, the decision as to whether or not to send extra information about a packet should be made depending on how important that packet is for reconstruction. We will describe this as dynamically adding redundancy.

4.1 Illustration of the Importance of Certain Packets

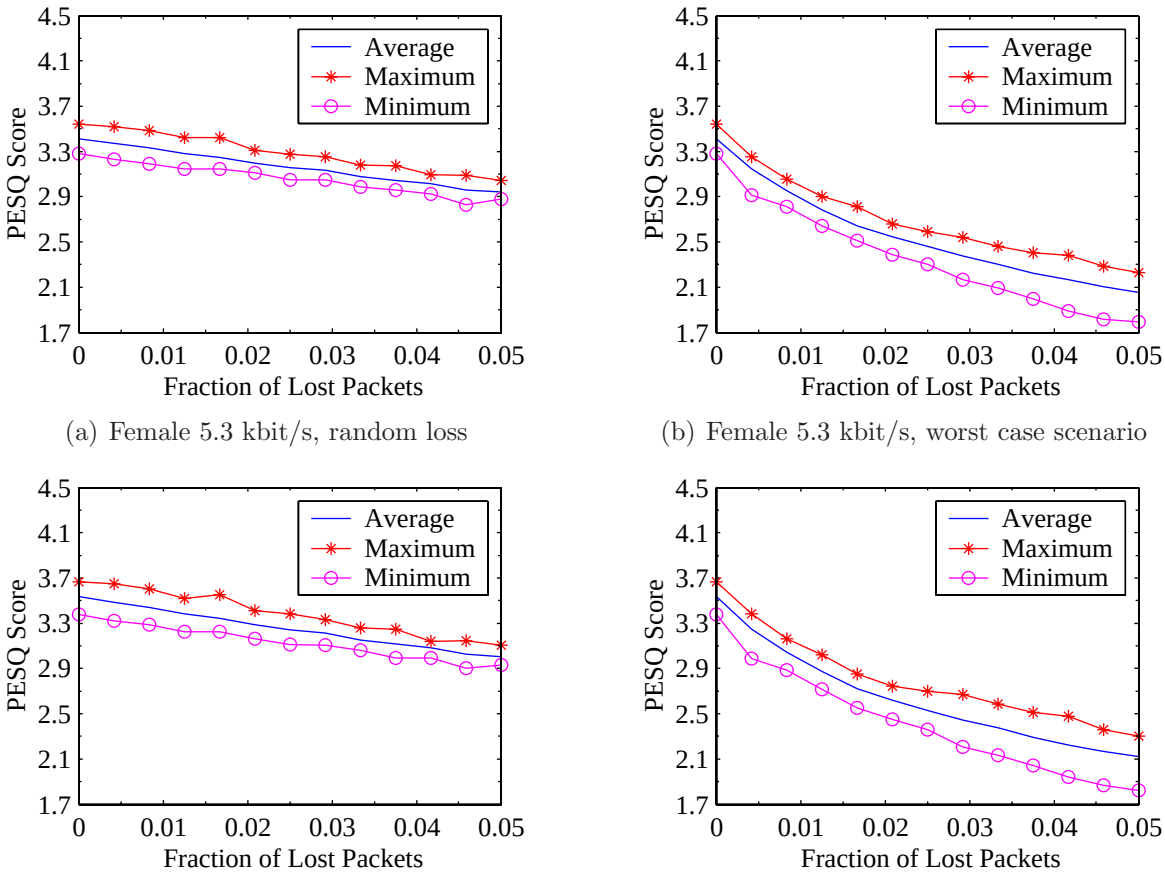
The whole idea of dynamic redundant information is based on the assumption that the packets are not equally important. To illustrate that certain packets are much more important than others, the following experiment is carried out. The same 11 female and 11 male speech files are used. In order to find the most important frames for packet loss concealment; one frame at a time is deemed to be lost for each file, the standard packet-loss-concealment scheme of G.723.1 is used and a PESQ score is found in each mode (6.3 kbit/s, 5.3 kbit/s). This gives N PESQ scores for a speech file with N frames. For each file and each mode, the PESQ scores are sorted from smallest to largest. The frames that correspond to the lowest PESQ scores are determined to be the most important frames. To

measure the performance of the packet-loss-concealment scheme of G.723.1 under a "worst-case-scenario", the location of the losses are selected from the most important frames as opposed to assigning them randomly as it was done in Chapter 3 to illustrate the performance of the packet-loss-concealment scheme of G.723.1 under random losses. Again an algorithm is used to avoid consecutive losses. The results are illustrated in Figs. 4.1 and 4.2. To illustrate the effect of important packets being lost, and to make it easier for the reader to compare the results, the figures that were obtained previously for random losses, in Chapter 3, are given on the left. The graphs obtained under worst-case-scenario losses (on the right) follow the same characteristics as those obtained under random loss (on the left) — PESQ scores decrease as the number of losses increases, the rate of decrease is not very different for any of the cases and the PESQ scores are better for male speech than for female speech and better at 6.3 kbit/s than at 5.3 kbit/s. However, the decrease rates for worst-case-scenario losses are much higher than those for random losses. Numerically speaking, on average, at 5.3 kbit/s, PESQ scores decrease by 39% for female speech (3.41 for no loss and 2.05 for 5% worst-case-scenario loss) and 37% for male speech (3.55 for no loss and 2.23 for 5% worst-case-scenario loss). At 6.3 kbit/s, they decrease by 39% for female speech (3.53 for no loss and 2.12 for 5% worst-case-scenario loss) and 37% for male speech (3.7 for no loss and 2.33 for 5% worst-case-scenario loss). Table 4.1 summarizes these results.

Table 4.1 PESQ scores for no loss and under 5% worst-case-scenario loss

Speech Signal	Coder Mode	No Loss	5% Worst-case-scenario Loss	Change
Female	5.3 kbit/s	3.41	2.05	39 %
Male	5.3 kbit/s	3.55	2.23	37 %
Female	6.3 kbit/s	3.53	2.12	39 %
Male	6.3 kbit/s	3.70	2.33	37 %

This experiment shows us that certain packets are indeed much more important than others. For example, for female speech at 5.3 kbit/s, the average PESQ score under 5% random losses is 2.94 whereas it is 2.05 for the same fraction of worst-case-scenario losses. A single packet loss at a critical frame can be quite audible. The second point in all the curves corresponds to $0.05/12 = 0.416\%$ losses and is equal to 1 lost packet for all the speech files. As it can be seen, the drop of the PESQ score from the no loss case is significantly higher for a packet loss at a critical frame than for a random packet loss. Therefore we need to



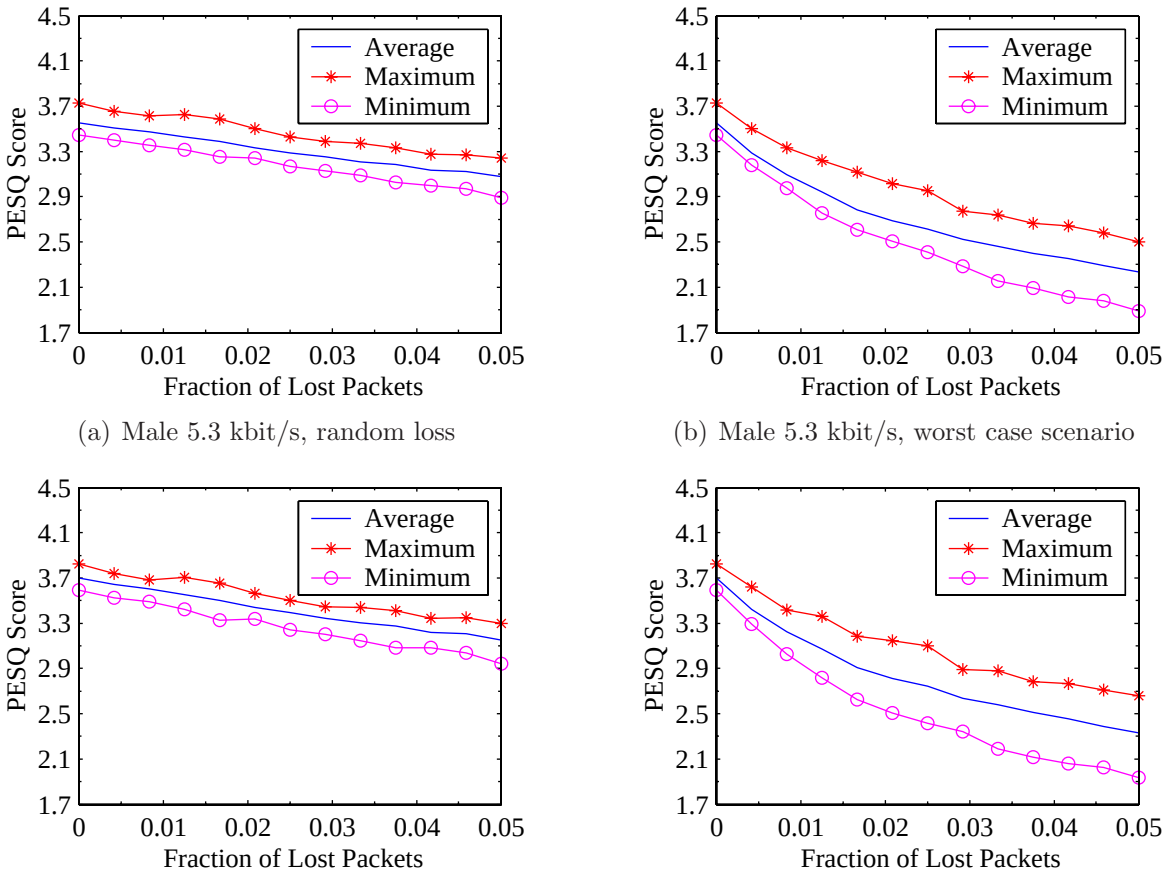
(a) Female 5.3 kbit/s, random loss

(b) Female 5.3 kbit/s, worst case scenario

(c) Female 6.3 kbit/s, random loss

(d) Female 6.3 kbit/s, worst case scenario

Fig. 4.1 Illustration of the importance of certain packets for female speech files



(a) Male 5.3 kbit/s, random loss

(b) Male 5.3 kbit/s, worst case scenario

(c) Male 6.3 kbit/s, random loss

(d) Male 6.3 kbit/s, worst case scenario

Fig. 4.2 Illustration of the importance of certain packets for male speech files

improve the packet loss concealment under worst-case-scenario losses.

The way we determine important packets here is not applicable for real-time processing. In real-time applications, we do not have the whole speech signal, hence we cannot use the method of sorting out all PESQ scores from the smallest to the largest to determine the important packets. We can solve this problem in two ways:

1. We determine a reference PESQ score for each file.

2. We find another way to determine the importance of each packet.

4.1.1 Defining a Reference PESQ Score

There are two ways to determine a reference PESQ score. The simplest one in terms of computation is predetermining a reference score for different cases. We know that G.723.1 has different performances for female and male speech, and at 5.3 kbit/s and 6.3 kbit/s. Therefore, we can define four different reference PESQ scores for these four cases, such that for each case, if the PESQ score that is found assuming that a packet is lost is smaller than the reference, we conclude that the packet in question is important. However, even for the same gender, PESQ scores vary for different speakers. Therefore defining a general reference PESQ score for each case, although easy in terms of computation since it is done only once, is not good enough.

G.723.1 has different performances for 6.3 kbit/s and 5.3 kbit/s because of algorithmic differences. Performance varies for female and male speech because female and male speech have different characteristics. However, neither all female speech nor all male speech have the same characteristics. This is why it is difficult to determine a general reference PESQ score according to the gender. Another way is to determine a reference PESQ score for each speaker. At the start of transmission for a given speaker, the first N frames might be used to determine a reference score for that speech. Since PESQ scores vary according to the characteristics of a speaker and remain the same as long as the speaker is the same, it is reasonable to assume that we can define a reference PESQ score for each speaker using the first N frames. From there on, we can use the reference PESQ score to determine the importance of a packet at the end of a block of K packets — in the coder, each packet is considered lost and packet loss concealment is carried out. Then a PESQ score is found for that block of the last K packets. The importance of the K^{th} packet in the block is determined by comparing the PESQ score with the reference PESQ score.

4.2 Importance of Different Aspects in Reproduction of Speech

Speech synthesis is carried out by modelling the vocal tract and excitation signal. When a packet is lost, a packet-loss-concealment scheme is used to regenerate the LP parameters and excitation parameters. As discussed in the previous section and shown in Figs. 4.1 and 4.2, past packets do not always contain enough information to reconstruct the data in certain packets, which causes the packet loss concealment to perform poorly. There are three possible reasons for this:

1. LP parameters cannot be regenerated and memory allocated for past LP parameters cannot be updated properly

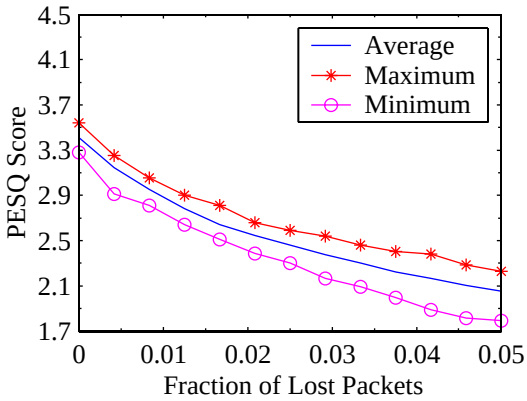
2. Excitation parameters cannot be reconstructed and memory allocated for past excitation parameters cannot be updated properly

3. Both of the first two cases

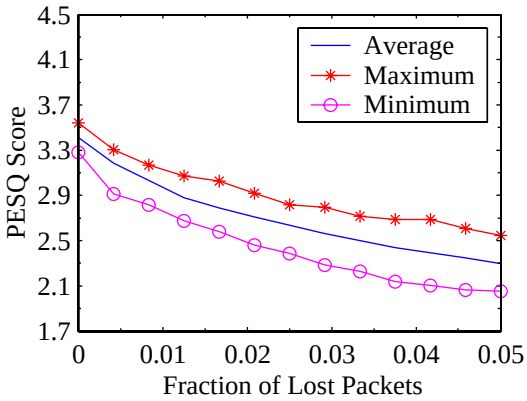
It is important that we understand where the packet-loss-concealment scheme fails, so that we can figure out not only the extra information we need to send to recover the important packets and update the memory, but also another way to determine if a packet is important or not.

4.2.1 LP Parameters

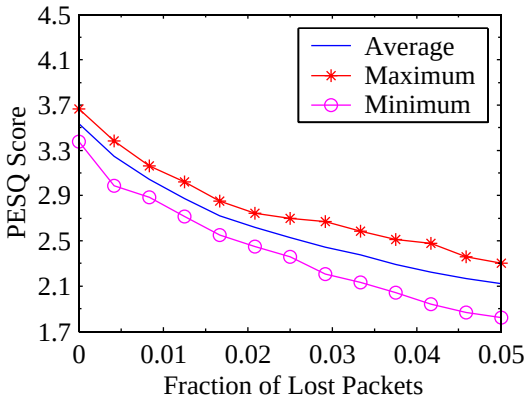
To figure out if it is the LP parameters that the packet loss concealment performs poorly to regenerate, LP parameters are sent as redundant information. The same 11 female and 11 male speech files are used. For the packet loss, the same loss pattern that was used to illustrate the importance of certain packets is used. The results are shown in Figs. 4.3 and 4.4. Parts a) and c) appeared earlier on Figs. 4.1 and 4.2 and they correspond to PESQ scores for important packets being lost. Graphs on the right correspond to PESQ scores obtained by sending LP parameters as redundant information and using them both in the reconstruction of the LP parameters of the lost packets and updating the LP memory. As can be observed, repeating LP parameters to use in the packet-loss-concealment procedure gives some improvement, but as will be shown in the following section, the improvement provided by repeating excitation parameters as opposed to LP parameters is much more.



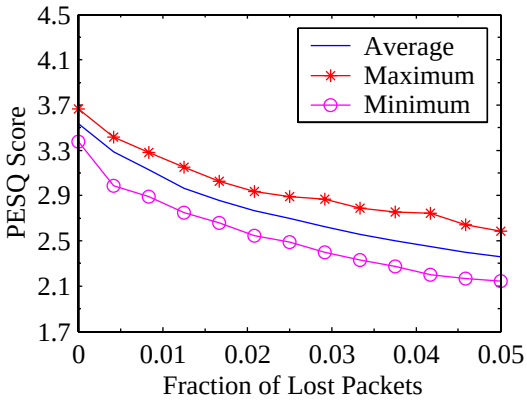
(a) Female 5.3 kbit/s, worst case scenario



(b) Female 5.3 kbit/s, worst case scenario, LP parameters are sent as extra information

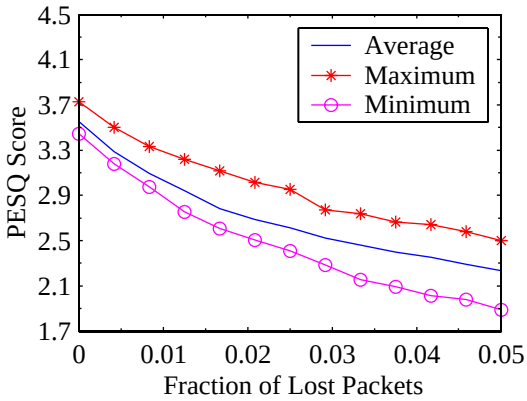


(c) Female 6.3 kbit/s, worst case scenario

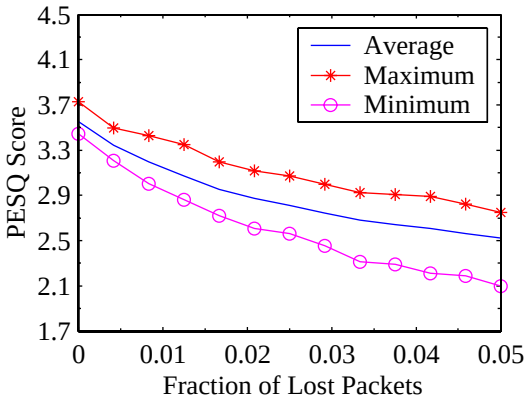


(d) Female 6.3 kbit/s, worst case scenario, LP parameters are sent as extra information

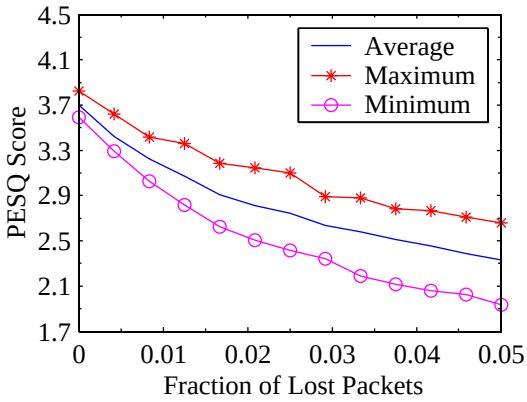
Fig. 4.3 Illustration of the effect of sending LP parameters as extra information for female speech files and using them both in the reconstruction of the lost LP parameters and in updating the LP memory



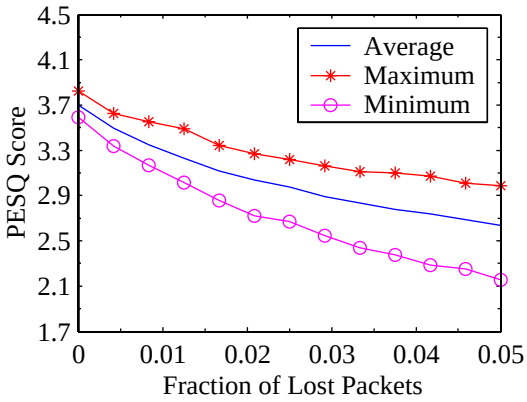
(a) Male 5.3 kbit/s, worst case scenario



(b) Male 5.3 kbit/s, worst case scenario, LP parameters are not lost



(c) Male 6.3 kbit/s, worst case scenario



(d) Male 6.3 kbit/s, worst case scenario, LP parameters are not lost

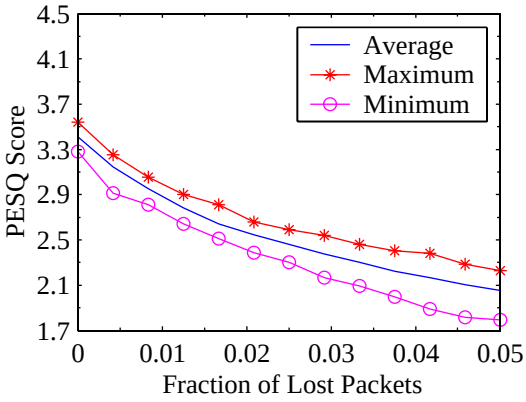
Fig. 4.4 Illustration of the effect of sending LP parameters as extra information for male speech files and using them both in the reconstruction of the lost LP parameters and in updating the LP memory

4.2.2 Excitation Parameters

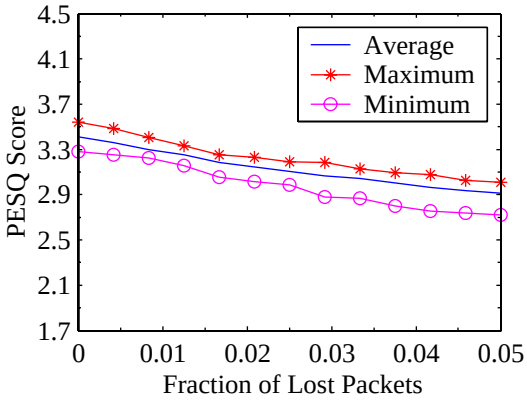
To find out if it is the excitation parameters that the packet loss concealment performs poorly to reconstruct, all the parameters related to the generation of excitation signal are sent as extra information. In other words, all the information in a packet is sent twice except for the LP parameters. The same 11 female and 11 male speech files are used. For the packet loss, the same loss pattern that was used to illustrate the importance of certain packets is used. The results are shown in Figs. 4.5 and 4.6. The importance of excitation parameters in the packet-loss-concealment scheme can easily be seen by comparing the graphs on the left to those on the right. Parts a) and c) appeared earlier on Figs. 4.1 and 4.2 and they correspond to PESQ scores for important packets being lost. The graphs on the right correspond to sending excitation parameters as redundant information. As can be seen, if excitation parameters of the important packets are sent as extra information, the performance of the packet-loss-concealment scheme improves significantly.

Figures 4.7 and 4.8 show the comparison of the improvement obtained by sending LP parameters as the redundant information to the improvement obtained by sending excitation parameters as the redundant information. The graphs on the left appeared before in Figs. 4.3 and 4.4. The graphs on the right appeared before in Figs. 4.5 and 4.5.

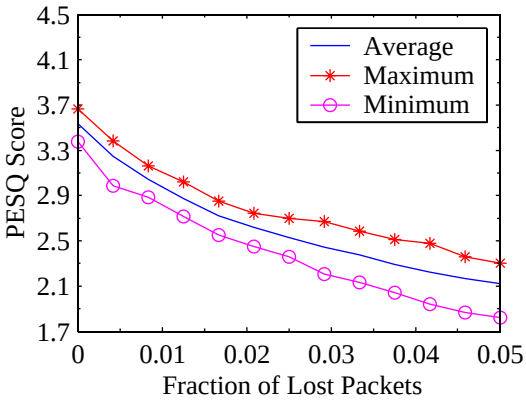
We had previously concluded that for certain cases packet loss concealment performs poorly to recover the lost data and to update the states. Observing Figs. 4.3, 4.4, 4.5, 4.6, 4.7 and 4.8, we can further conclude that for these important packets, it is not the LP parameters but the excitation parameters that the packet loss concealment reconstructs poorly. This is in line with the proposition that LP parameters do not change rapidly from frame to frame and they can be more easily regenerated using past LP parameters. Hence it is not necessary to send them as extra information. On the other hand, sending the excitation parameters of the most important packets as extra information improves the packet loss concealment significantly. Therefore we can conclude that we must consider sending the excitation parameters of the important packets as redundant information. As proposed earlier, we can further conclude that excitation parameters can be used to determine if a packet is important.



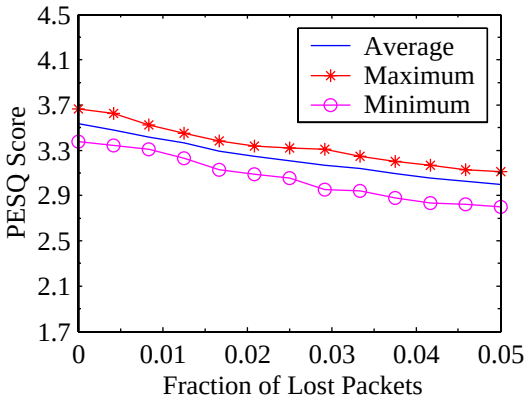
(a) Female 5.3 kbit/s, worst case scenario



(b) Female 5.3 kbit/s, worst case scenario, excitation parameters are sent as extra information

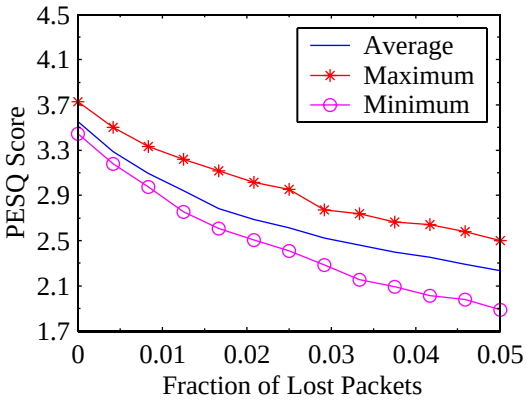


(c) Female 6.3 kbit/s, worst case scenario

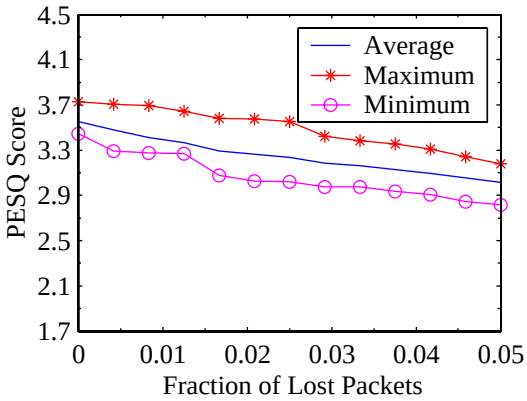


(d) Female 6.3 kbit/s, worst case scenario, excitation parameters are sent as extra information

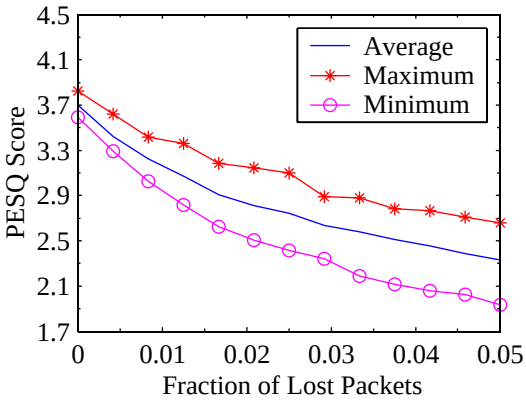
Fig. 4.5 Illustration of the effect of sending excitation parameters as extra information for female speech files and using them both in the reconstruction of the lost excitation parameters and in updating the excitation memory



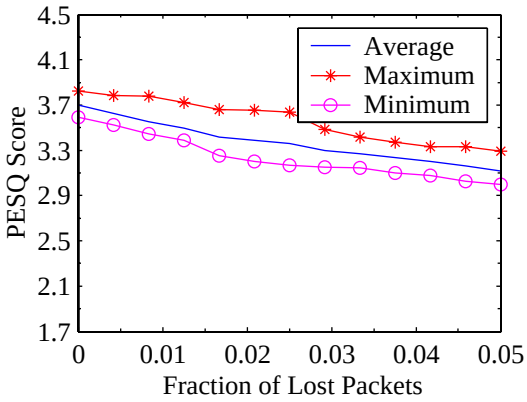
(a) Male 5.3 kbit/s, worst case scenario



(b) Male 5.3 kbit/s, worst case scenario, excitation parameters are sent as extra information

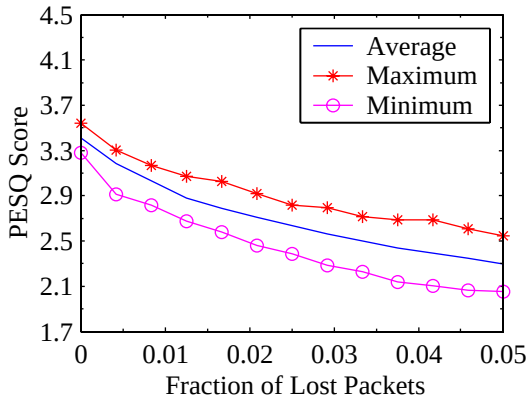


(c) Male 6.3 kbit/s, worst case scenario

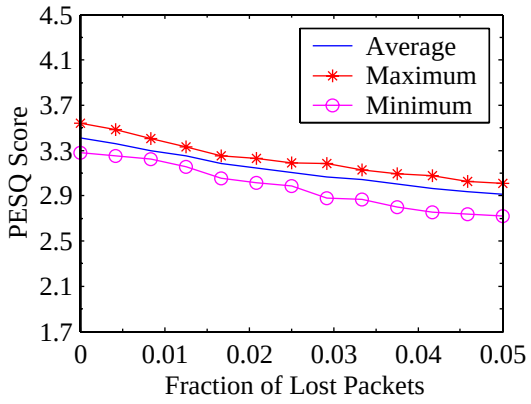


(d) Male 6.3 kbit/s, worst case scenario, excitation parameters are sent as extra information

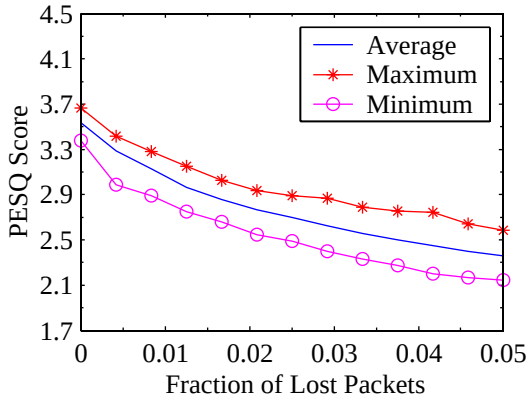
Fig. 4.6 Illustration of the effect of sending excitation parameters as extra information for male speech files and using them both in the reconstruction of the lost excitation parameters and in updating the excitation memory



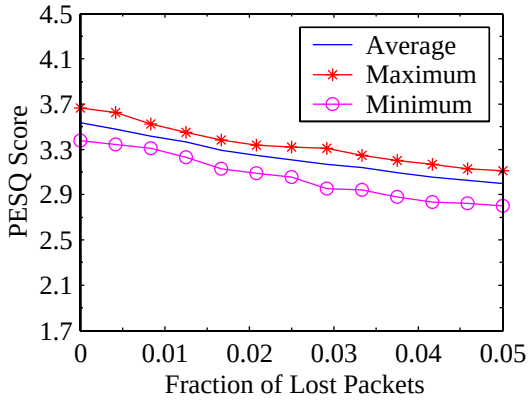
(a) Female 5.3 kbit/s, worst case scenario, LP parameters are sent as extra information



(b) Female 5.3 kbit/s, worst case scenario, excitation parameters are sent as extra information

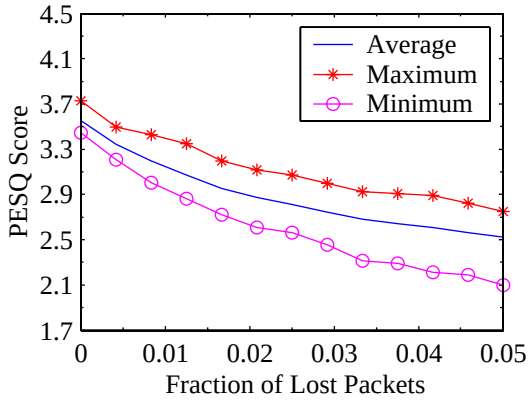


(c) Female 6.3 kbit/s, worst case scenario, LP parameters are sent as extra information

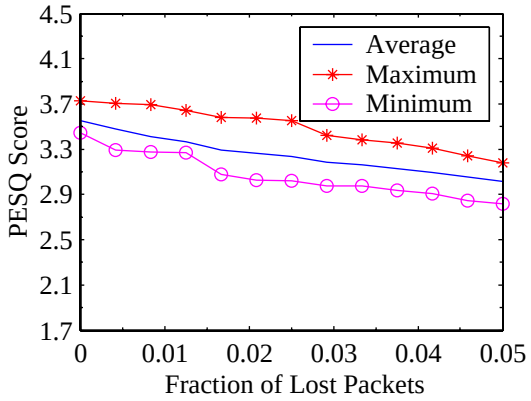


(d) Female 6.3 kbit/s, worst case scenario, excitation parameters are sent as extra information

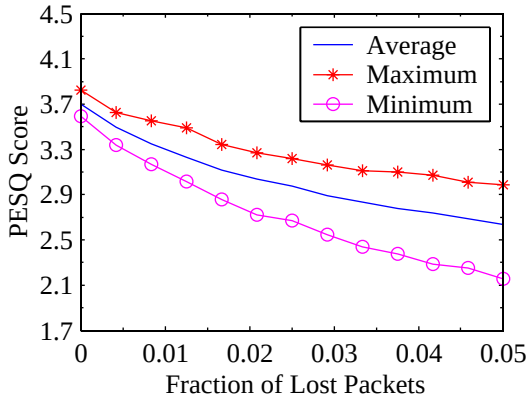
Fig. 4.7 Comparison of sending LP parameters to sending excitation parameters as extra information for female speech files



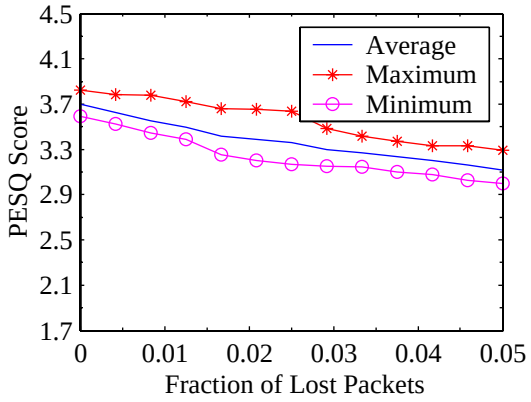
(a) Male 5.3 kbit/s, worst case scenario, LP parameters are sent as extra information



(b) Male 5.3 kbit/s, worst case scenario, excitation parameters are sent as extra information



(c) Male 6.3 kbit/s, worst case scenario, LP parameters are sent as extra information



(d) Male 6.3 kbit/s, worst case scenario, excitation parameters are sent as extra information

Fig. 4.8 Comparison of sending LP parameters to sending excitation parameters as extra information for male speech files

4.3 Using Excitation Parameters in Packet Loss Concealment

We observed that excitation parameters play a crucial role in the decoding process and that the loss of the excitation parameters of certain packets cannot be concealed using the packet-loss-concealment scheme of G.723.1. As a result of this observation, we concluded that we must consider sending excitation parameters as extra information for important packets. However, we must further determine how we should use this extra information. The excitation parameters which are sent as redundant information can be used in two different ways:

1. The excitation parameters that are sent as extra information can be used in the reconstruction of the excitation parameters of the lost frame (as a consequence, the states of the subsequent frame are updated).

2. The packet-loss-concealment scheme can be used to generate an excitation signal for the lost packet and the extra information is only used to update the excitation memory.

There are two ways to send extra information. It can be sent with the previous packet or with the next packet. If the purpose is to use the excitation parameters in the regeneration of the excitation parameters of the lost frame, then regardless of whether extra information is sent with the previous or the next packet, a one-frame-delay is introduced. If extra information is sent with the previous packet, we introduce this delay in the coder since to add extra information to a packet about the next frame, we must wait for that next frame. If, on the other hand, this extra information is sent with the next packet, the delay is introduced in the decoder, since in case of a loss, we must wait for the next packet to use the extra information in the packet-loss-concealment process. Therefore, although using the extra information in the reconstruction of the lost excitation parameters is a better concealment method, it has a disadvantage — it introduces a one frame delay.

The second method does not use the redundant information in the reconstruction of the lost excitation parameters but only in the decoding process of the next packet, since the excitation memory is only used for the next packet. Hence, for the second method it becomes important whether we send the extra information with the previous or the next packet. Sending extra information with the previous packet, regardless of whether we use that information to reconstruct the lost excitation parameters or only to update the

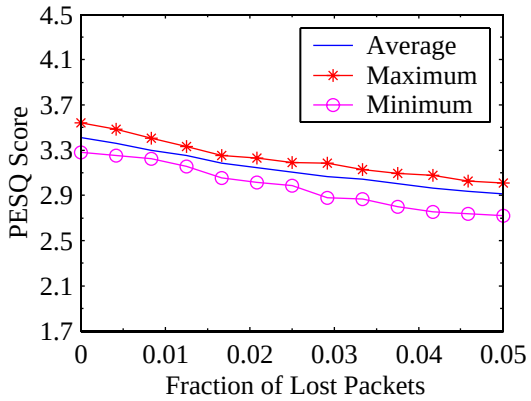
states, introduces a delay. If, on the other hand, the extra information is intended to be used only to update the states, then sending this information with the next packet does not introduce any additional delay. Hence, if the extra information is sent with the next packet, the second method has the advantage of not introducing an additional delay.

There is a trade-off between delay and quality between the two methods. The following experiment shows the improvements that these two methods provide in the packet loss concealment. The same 11 female and 11 male speech files are used. For the packet loss, the same loss pattern that was used to illustrate the importance of certain packets was used. The results are shown in Figs. 4.9 and 4.10. The importance of using excitation parameters in the decoding process can easily be seen by comparing the graphs on the left to those on the right. Parts a) and c) appeared earlier on Figs. 4.5 and 4.6 and they correspond to sending excitation parameters as redundant information and using them to reconstruct the lost excitation parameters — as a consequence, the excitation memory is updated. The graphs on the right correspond to sending excitation parameters as redundant information, but using them only to update the excitation memory. As can be seen, using excitation parameters only to update the memory provides only a small improvement in the packet loss concealment.

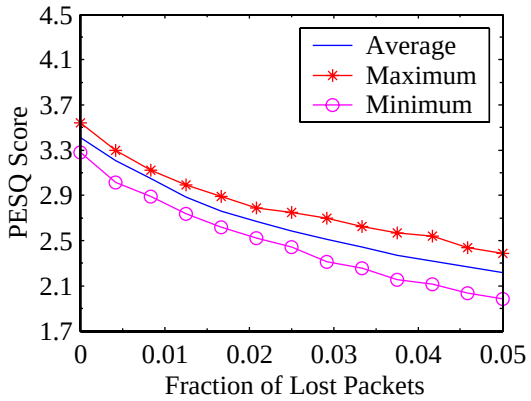
This experiment shows that the excitation parameters of the important packets should be used in the reconstruction of the excitation parameters of the lost frame, in which case the states of the subsequent frame are updated consequently.

4.4 Using Excitation Parameters to Determine Packet Importance

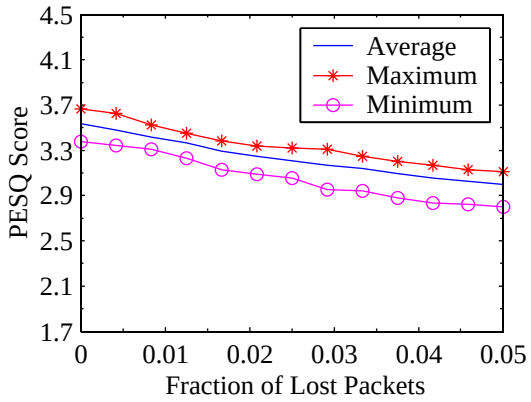
As we concluded in the previous section, observing the excitation parameters is the key to determining the important packets. To figure out why the excitation parameters corresponding to certain frames are more important than others, we compared the excitation signals of the important packets to the excitation signals of the frames right before them and excitation signals generated by the packet-loss-concealment scheme when they are lost. An example is given in Fig. 4.11. As can be seen in Fig. 4.11, the excitation signal of an important frame corresponds to a voiced frame. The previous frame's excitation signal, on the other hand, indicates that the frame is unvoiced. We can also observe that the excitation signal generated by the packet-loss-concealment scheme of G.723.1 resembles the



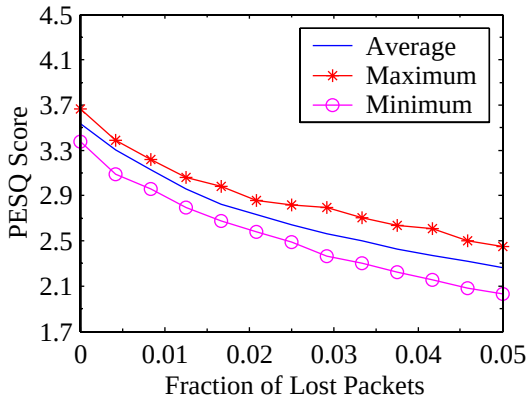
(a) Female 5.3 kbit/s, worst case scenario, excitation parameters are sent as extra information



(b) Female 5.3 kbit/s, worst case scenario, only the excitation memory is updated

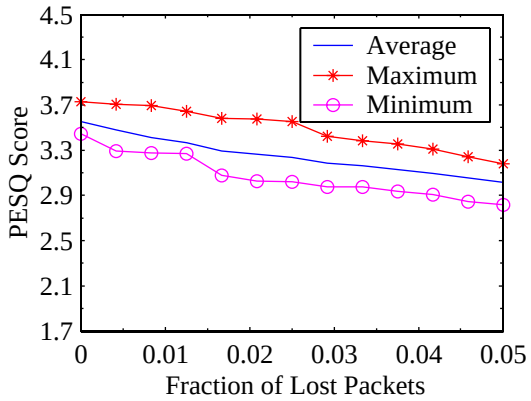


(c) Female 6.3 kbit/s, worst case scenario, excitation parameters are sent as extra information

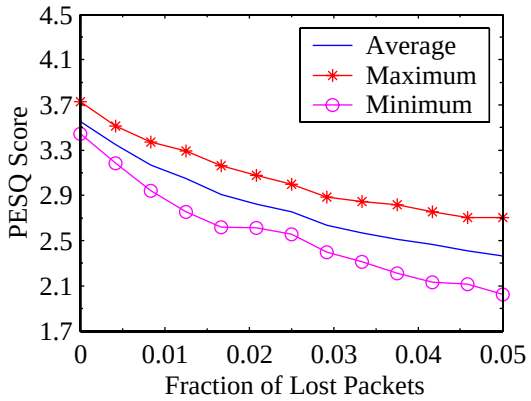


(d) Female 6.3 kbit/s, worst case scenario, only the excitation memory is updated

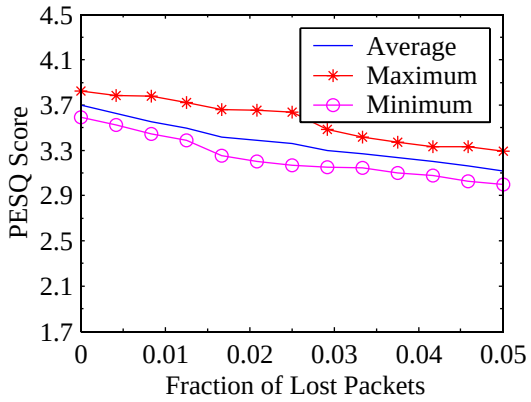
Fig. 4.9 Comparison of using excitation parameters in the reconstruction of the lost excitation parameters to using them only to update the states for female speech files



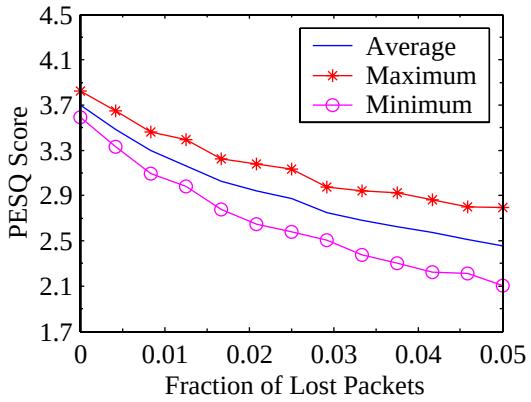
(a) Male 5.3 kbit/s, worst case scenario, excitation parameters are sent as extra information



(b) Male 5.3 kbit/s, worst case scenario, only the excitation memory is updated

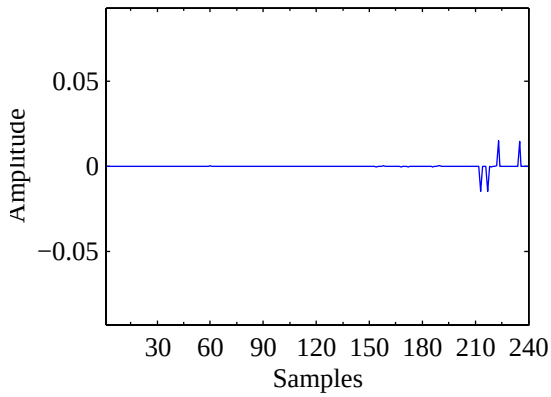


(c) Male 6.3 kbit/s, worst case scenario, excitation parameters are sent as extra information

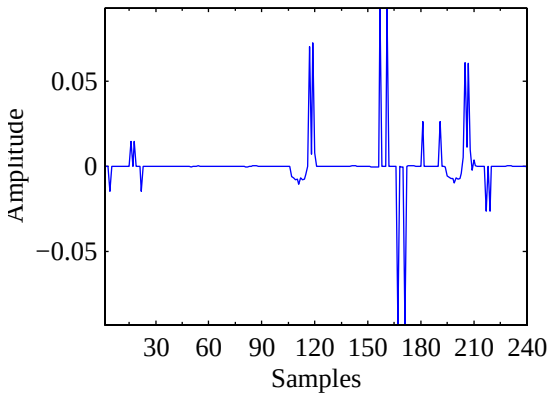


(d) Male 6.3 kbit/s, worst case scenario, only the excitation memory is updated

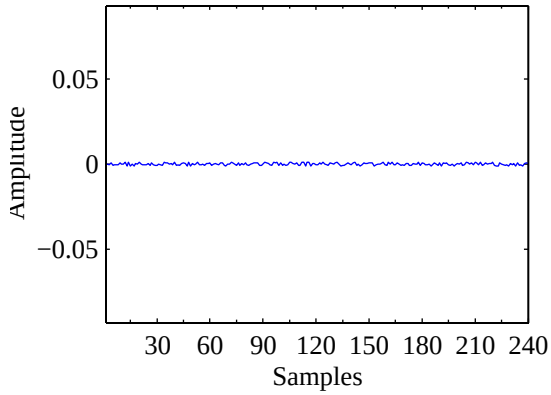
Fig. 4.10 Comparison of using excitation parameters in the reconstruction of the lost excitation parameters to using them only to update the states for male speech files



(a) Excitation signal of the previous frame



(b) Excitation signal of an important frame



(c) Excitation signal generated by the G.723.1 packet-loss-concealment scheme to replace the excitation signal of the important frame given in part (b)

Fig. 4.11 Comparison of excitation signal of an important frame with the excitation signal of the previous frame and the excitation signal generated by the packet-loss-concealment scheme when it is the only lost packet

excitation signal of the previous frame. The packet-loss-concealment scheme of G.723.1 uses past excitation parameters to regenerate the excitation parameters of a lost frame. In the specific case of a voiced frame following an unvoiced one, when the voiced frame is lost, it is inevitable for the packet-loss-concealment scheme to generate an excitation signal of an unvoiced frame.

We had previously concluded that certain packets are much more important than others and when they are lost, their excitation parameters cannot be adequately regenerated using the packet-loss-concealment scheme of G.723.1 and that the excitation parameters of these packets should be sent as extra information to improve the packet loss concealment. This experiment shows that the packets that were determined to be important are voiced frames following an unvoiced frame.

This experiment reinforces the proposition that excitation parameters of the important frames should be sent as extra information and be used in the reconstruction of the lost excitation parameters. This experiment also gives the answer to the question as to how we can determine if a packet is important or not: observing the excitation signal. There are two things to do:

1. If the VAD/CNG (voice activity detection and comfort noise generation) option is activated, we can easily figure out the important packets by looking at the coding mode. If, in the coding process, the SID (silence insertion description, explained in chapter 2) or NULL mode is followed by an ACELP or MP-MLQ mode, we can conclude that a voiced frame is following an unvoiced frame and we can determine that this voiced frame is important and its excitation parameters should be sent as extra information with the following packet

2. By using a simple algorithm we can compare the excitation signals of two consecutive frames and see if a voiced frame is following an unvoiced one.

It is easy to see that the main difference between the excitation signal of an important packet and that of the previous one is the lack of peaks in the latter. Figure 4.11 shows the excitation signal of a frame that was determined to be the most important one as a result of the experiment described earlier in the chapter. Therefore the difference between the excitation signal of an important frame and that of the previous frame is not always as significant. However, it is observed that the energy of the peaks of excitation signals of

important frames is significantly larger than the energy of the peaks of excitation signals of the previous frames.

We have developed a method to determine the importance of packets. We calculate the ratios of the average peak magnitude and the rms of the excitation signal of a frame to those of the previous frame. The frame is determined to be important if either one of the ratios is greater than 5. We tested this method on every speech file (11 female and 11 male speech files) and for each mode (5.3 kbit/s and 6.3 kbit/s). The results are given in Table 4.2. These results can be summarized as follows: For female and male speech files and

Table 4.2 Ratio of the number of the important frames to the total number of frames

File number	Female 5.3 kbit/s	Female 6.3 kbit/s	Male 5.3 kbit/s	Male 6.3 kbit/s
1	36/314	35/314	41/304	36/304
2	34/324	31/324	48/320	48/320
3	32/298	31/298	25/265	25/265
4	29/307	30/307	34/331	32/337
5	35/317	32/317	34/278	31/278
6	40/276	39/276	35/311	33/311
7	31/360	33/360	26/315	27/315
8	47/361	49/361	40/389	39/389
9	37/325	34/325	36/278	35/278
10	49/356	48/356	38/302	37/302
11	29/295	28/295	27/240	24/240

at 5.3 kbit/s and 6.3 kbit/s, on average, 11% of the packets are determined to be important. As we concluded before, the excitation parameters for the packets that are determined to be important must be sent as extra information. As shown before in Chapter 2, to code the excitation signal of a frame, 165 bits out of the 189 bits are used at 6.3 kbit/s and 134 out of the 158 bits are used at 5.3 kbit/s. This means that for 6.3 kbit/s we must send extra 165 bits for 11% of the packets and for 5.3 kbit/s we must send 134 bits. This makes an average bit-rate of 6.9 kbit/s $((189 + 165 \times 0.11) \div 30 = 6.9)$ for the MP-MLQ mode and an average bit-rate of 5.8 kbit/s $((158 + 134 \times 0.11) \div 30 = 5.8)$ for the ACELP mode. If we send excitation parameters twice for each frame as opposed to sending them for the most important packets, it makes a fixed bit-rate of 11.8 kbit/s $((189 + 165) \div 30 = 11.8)$ for the MP-MLQ mode and a fixed bit-rate of 9.7 kbit/s $((158 + 134) \div 30 = 9.7)$ for the

ACELP mode. The bandwidth reduction provided by sending extra information only for the important packets is significant.

4.4.1 New Redundancy-Based Packet-Loss-Concealment Scheme

If the excitation parameters of a frame are determined to be important for packet loss concealment, they are sent with the subsequent packet. Sending excitation parameters twice for the important packets results in an average bit-rate of 6.9 kbit/s for MP-MLQ and 5.8 kbit/s for ACELP. The additional bandwidth introduced due to adding redundancy only for the important packets is significantly smaller than adding redundancy for each and every packet (11.8 kbit/s for MP-MLQ and 9.7 kbit/s for ACELP). Hence, the method provides an improved packet loss concealment at a modest increase in the average bit-rate. Table 4.3 summarizes these results.

Table 4.3 The comparison of sending extra information for important packets to sending them for every packet in terms of bit-rate

Coder Mode	Standard PLC (G.723.1)	EXC par. (each frame)	EXC par. (imp. frames)
ACELP	5.3 kbit/s	9.7 kbit/s	5.8 kbit/s
MP-MLQ	6.3 kbit/s	11.8 kbit/s	6.9 kbit/s

With the improvement that this method provides on the packet loss concealment, the effect of the worst-case-scenario losses is reduced to that of random losses. In other words, with the improvement provided by the proposed method, 5% worst-case-scenario losses can be tolerated.

4.5 Chapter Summary

In this chapter, we first showed that certain packets are much more important for packet loss concealment than others. We suggested that packet loss concealment should be improved for these important packets. We showed that sending the excitation parameters of the important packets provide a significantly better improvement in the packet loss concealment than sending the LP parameters. Therefore, we concluded that the excitation parameters of the important packets should be sent as extra information.

Then we compared the improvement of using the excitation parameters that are sent as extra information only in updating the excitation memory to using them in regenerating the lost excitation parameters, in which case the excitation memory is updated as a consequence. We observed that the improvement of using the extra information to regenerate the lost excitation parameters is significantly better than using it just to update the states. Hence we concluded that the excitation parameters sent as extra information should be used in the reconstruction of the lost excitation parameters.

We then observed that the frames that were determined to be important were voiced frames following unvoiced frames. We proposed an algorithm to determine the importance of the packets and we showed that using our algorithm, on average, 11% of the packets are determined to be important, which results in an 11% increase in the bit-rate on average. This increase is much lower compared to the increase that would be obtained by duplicating the excitation parameters of all the frames. We further proposed that with the improvement that this method provides on the packet loss concealment, the effect of the worst-case-scenario losses is reduced to that of random losses, which means that 5% worst-case-scenario losses can be tolerated.

Chapter 5

Conclusion

5.1 Summary and Discussion of Results

Modern speech coders try to take advantage of redundancies found in speech signals in order to code speech signals with very few bits while keeping the quality sufficiently high. As a result of this, decoding of each packet becomes dependent on the successful transmission and decoding of the previous packets. In other words, in the decoding process of each packet, past information gained from the decoding of previous packets is used. The dependence on past frames to decode the current frame introduces a concept of coder state. Using past information to decode a frame, although providing lower bit-rates, causes problems in an Internet environment due to variable delays experienced — packets do not always arrive in time for play-out and are thus considered lost. When a packet is lost, coder states cannot be updated properly and due to state synchronization problems, the effect of a lost packet propagates to subsequent packets.

Considerable research has been done to deal with the packet loss problem and several packet-loss-concealment algorithms have been proposed. We discussed these algorithms in two categories: receiver-based schemes and sender-receiver-based schemes. Receiver-based schemes are simple, yet do not have a very good performance, because they rely on the assumption that packet losses do not occur very frequently or consecutively. However, this assumption does not hold for the Internet. We discussed the sender-receiver-based schemes in three categories: priority-based schemes, redundancy-based schemes and interleaving-based schemes. Priority-based schemes assign priorities to packets according to their importance assuming that there is a network which supports dropping packets according to

their preassigned priorities, which again does not hold for the Internet. Redundancy-based schemes add extra information about each packet to the previous or subsequent packet, which is used to regenerate the waveform that a lost packet corresponds to. As a result, they increase the bit-rate of the coder. Interleaving-based schemes, in contrast to redundancy-based schemes, do not increase the bit-rate. They deal with the problem of packet loss by distributing the loss of the information in a packet over several packets. Interleaving relies on the assumption that there is correlation between the parameters that are sent in each packet. This assumption holds for waveform coders, since there is correlation between speech samples. However, modern coders do not send the speech itself but parameters that are used to reproduce it with minimum error, by matching the waveform as much as possible. There is correlation between the LP parameters, but not between excitation parameters. Therefore, most successful interleaving methods also end up sending excitation parameters twice as redundant information. Since excitation parameters comprise the significant portion of the data sent in a packet, we concluded that redundancy based schemes should be preferred over interleaving-based schemes.

Most of the current redundancy-based schemes send extra information for each and every packet, regardless of their importance. In this thesis we focused on defining the importance of each packet and sending extra information for the important packets, thereby causing a smaller increase in the bit-rate. We showed that certain packets are more important than others. We illustrated that it does not give much improvement to send LP parameters. This is in line with the proposition that LP parameters are correlated, and hence can be recovered using LP parameters of previous packets. On the other hand, we showed that sending excitation parameters as the redundant information does make a significant improvement in the quality of the decoded speech. Therefore, we concluded that it is not necessary to send extra information for every packet but only for the important ones, and that the extra information must be excitation parameters, not LP parameters. We then examined how the excitation parameters can be used. We discussed two possibilities:

1. Excitation parameters are used only to update the coder states so that the effect of the lost packet does not propagate to the subsequent packets.

2. Excitation parameters are used to reconstruct the lost excitation parameters — as a consequence, the coder states are updated.

To use the excitation parameters that are sent as extra information in recovering the exci-

tation parameters of a lost packet, one must wait for the arrival of the next packet, which introduces an additional delay. Using excitation parameters only to update the states avoids this additional delay. We showed that there is a significant difference in terms of the final quality of the speech between the two methods, therefore it is necessary that the excitation parameters be used in the reconstruction of the lost excitation parameters despite the additional delay. Finally, we tried to find a way to determine if a packet is important or not. We illustrated that the most important packets are those that correspond to voiced sections of a speech signal following packets that correspond to unvoiced sections. We proposed two methods to determine if a packet is important:

1. A reference PESQ score is defined using the first few packets. For subsequent packets, a PESQ score is determined assuming that the packet in question is lost. This resultant PESQ score is then compared to the reference score to determine the importance of that packet.

2. In the coder, the excitation signal of each packet is compared with that of the preceding one. Peaks of the excitation signals of a packet corresponding to voiced sections of speech signals are significantly larger that those corresponding to unvoiced sections of speech signals. Therefore comparing the peaks, it is determined if the packet in question is a voiced one and is following an unvoiced packet, in which case it is determined to be important.

We showed that as a result of the second method, 11% of the packets are determined to be important on average. Therefore, sending excitation parameters twice for the important packets results in 6.9 kbit/s for MP-MLQ and 5.8 kbit/s for ACELP. The additional bandwidth introduced due to adding redundancy only for the important packets is significantly smaller compared to adding redundancy for each and every packet (11.8 kbit/s for MP-MLQ and 9.7 kbit/s for ACELP). Hence, we concluded that the improvement that this method provides is significant.

For test purposes, 11 female and 11 male speech files were used and they were coded and decoded for hundreds of different cases. It was not practical to perform a subjective test with real listeners for all the test files. Therefore PESQ was used as a subjective test tool. However, the conclusions that have been made based on PESQ scores were verified with subjective listening tests made on a sampling of the test conditions.

5.2 Future Work

5.2.1 Consecutive Losses

During the tests, both for random and for worst-case-scenario losses, consecutive losses are avoided not to overemphasize the importance of sending redundant information to improve packet loss concealment. This research can be expanded to cover consecutive losses, as well.

5.2.2 Improving the Algorithm used to Determine Importance

As we recall from Chapter 4, we followed the following method to illustrate the importance of certain packets: for each speech file, we considered each frame as if it were lost, and we found a PESQ score. We then sorted out these PESQ scores from smallest to largest (the number of the PESQ scores sorted out for a speech file is equal to the number of frames in that speech file). The packet that the smallest PESQ score corresponded to was determined to be the most important packet. Hence the sorted out list, from top to bottom, was the list of the frames sorted out according to their importance. Maximum amount of loss was determined to be 5%. For each step, which was also defined in percentages, the number of losses were determined. Then, according to the predetermined number of losses for each step, the packets to be considered lost were determined by referring to the list. Then the effect of worst-case-scenario losses was illustrated.

In the algorithm we use, a packet is determined to be important if any one of the following two conditions are met.

1. Average energy of the peaks of the excitation signal of the frame in question is at least 5 times larger than that of the preceding frame's.

2. The rms of the total energy of the excitation signal of the frame in question is at least 5 times larger than that of the preceding frame's.

The number 5 is determined in such a way that the packets to be considered lost would be determined to be important when the algorithm is applied. However, we never actually made a test to see if all of those packets were actually important. For example; P_1 being the most important packet, P_{10} being the 10^{th} most important packet, we know that the loss of the packets P_1 to P_5 has a very negative effect and we know that the effect of the loss

of the packets P_1 to P_{10} is worse; however, we never actually made a test to see the effect of the loss of packets P_6 to P_{10}. It might not be necessary to send the excitation parameters of packets P_6 to P_{10} as extra information. In other words, we did not test the limits of the algorithm. If we use a bigger number than 5 to determine the important packets, there will be fewer packets determined to be important. We said that on average 11% of the packets are determined to be important. Testing the limits of the algorithm and finding the best coefficient, this 11% could possibly be decreased.

References

[1] W. C. Chu, *Speech Coding Algorithms - Foundation and Evoulution of Standardized Coders*. Wiley, 2003.

[2] D. O'Shaughnessy, *Speech Communications: Human and Machine*. IEEE Press, 2000.

[3] ITU-T, Telecommunication Standardization Sector of ITU, *Pulse Code Modulation (PCM) of Voice Frequencies, ITU-T Recommendation G.711*, Nov. 1988.

[4] ITU-T, Telecommunication Standardization Sector of ITU, *40, 32, 24, 16 kbit/s adaptive diferential pulse code modulation (ADPCM), ITU-T Recommendation G.726*, Dec. 1990.

[5] U. Black, *Voice Over IP*. Prentice Hall, 2000.

[6] D. G. Manolakis, V. K. Ingle, and S. M. Kogon, *Statistical and Adaptive Signal Processing*. McGraw-Hill, 2000.

[7] ITU-T, Telecommunication Standardization Sector of ITU, *Coding of speech at 8 kbit/s using Conjugate-Structure Algebraic-Code-Excited Linear-Prediction CS-ACELP, ITU-T Recommendation G.729*, Mar. 1996.

[8] ITU-T, Telecommunication Standardization Sector of ITU, *Dual Rate Speech Coder for Multimedia Communications Transmitting at 5.3 and 6.3 kbit/s, ITU-T Recommendation G.723.1*, Mar. 1996.

[9] M. R. Schroeder and B. S. Atal, "Code-excited linear prediction (CELP): High-quality speech at very low bit rates," *Proc. IEEE Int. Conf. Acoust. Speech Signal Processing*, pp. 937–940, Mar. 1985.

[10] J.-P. Adoul, P. Mabilleau, M. Delprat, and S. Morissette, "Fast CELP coding based on algebraic codes," *Proc. IEEE Int. Conf. Acoust. Speech Signal Processing*, 1987.

[11] ITU-T, Telecommunication Standardization Sector of ITU, *Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs*, Feb. 2001.

[12] B. W. Wah, X. Su, and D. Lin, "A survey of error-concealmant schemes for real-time audio and video transmissions over the internet," *Proc. IEEE Int. Symposium Multimedia Software Engineering*, pp. 17–24, Dec. 2000.

[13] A. Shah, S. Atungsiri, A. Kondoz, and B. Evans, "Lossy multiplexing of low bit rate speech in thin route telephony," *IEE Electronic Letters*, vol. 32, pp. 95–97, Jan. 1996.

[14] J. Suzuki and M. Taka, "Missing packet recovery techniques for low bit-rate coded speech," *IEEE J. Select. Areas Commun.*, vol. 7, pp. 707–717, June 1989.

[15] R. C. F. Tucker and J. E. Flood, "Optimizing the performance of packet-switch speech," *IEEE Conf. Digital Processing of Signals Commun.*, pp. 227–234, Apr. 1985.

[16] V. Hardman, M. A. Sasse, M. Handley, and A. Watson, "Reliable audio for use over the internet," *Proc. INET*, June 1995.

[17] M. Yong, "Study of voice packet reconstruction methods applied to CELP speech coding," *Proc. IEEE Int. Conf. Acoust. Speech Signal Processing*, vol. 2, pp. 125–128, Mar. 1992.

[18] M. M. Lara-Barron and G. Lockhart, "Packet-based embedded encoding for transmission of low-bit-rate-encoded speech in packet networks," *Commun. Speech Vision, IEE Proc. 1*, vol. 139, pp. 482–487, Oct. 1992.

[19] G. Lockhart and M. M. Lara-Barron, "Implementation of packet-based encoding schemes for speech transmission," *Int. Conf. Digital Processing of Signals Commun.*, pp. 326–330, Sept. 1991.

[20] T. J. Kostas, M. S. Borella, I. Sidhu, G. M. Schuster, J. Grabiec, and J. Mahler, "Real-time voice over packet-switched networks," *IEEE Network*, vol. 12, Jan. 1998.

[21] N. Shacham, "Packet recovery and error correction in high-speed wide-area networks," *IEEE Military Commun. Conf.*, vol. 2, pp. 551–557, Oct. 1989.

[22] N. Shacham and P. McKenney, "Packet recovery in high-speed networks using coding and buffer management," *Proc. IEEE INFOCOM*, vol. 1, pp. 124–131, June 1990.

[23] H. Sanneck, "Concelament of lost speech packets using adaptive packetization," *IEEE Int. Conf. Multimedia Computing and Systems*, pp. 140–149, June 1998.

[24] R. A. Valenzuela and C. N. Animalu, "A new voice-packet reconstruction technique," *Proc. IEEE Int. Conf. Acoust. Speech Signal Processing*, vol. 2, pp. 1334–1336, May 1989.

[25] B. W. Wah and D. Lin, "LSP-based multiple-description coding for real-time low bit-rate voice transmission," *Proc. IEEE Int. Conf. Multimedia and Expo*, vol. 2, pp. 597–600, Aug. 2002.

[26] N. Jayant, "Effects of packet losses on waveform coded speech," *Proc. Int. Conf. Computer Commun.*, pp. 275–280, Oct. 1980.

[27] P. Gournay, F. Rousseau, and R. Lefebvre, "Improved packet loss recovery using late frames for prediction-based speech coders," *Proc. IEEE Int. Conf. Acoust. Speech Signal Processing*, vol. 1, pp. 108–111, Apr. 2003.