We are IntechOpen, the world's leading publisher of Open Access books
Built by scientists, for scientists

4,800
Open access books available

122,000
International authors and editors

135M
Downloads

154
Countries delivered to

Our authors are among the

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

BOOK CITATION INDEX
CLARIVATE ANALYTICS
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Enhanced VoIP by Signal Reconstruction and Voice Quality Assessment

Filipe Neves[1,2], Salviano Soares[3,4], Pedro Assunção[1,2] and Filipe Tavares[5]
[1]*Instituto Politécnico de Leiria*
[2]*Instituto de Telecomunicações*
[3]*Universidade de Trás-os-Montes e Alto Douro*
[4]*Instituto de Engenharia Electrónica e Telemática de Aveiro*
[5]*Portugal Telecom Inovação*
Portugal

1. Introduction

The Internet and its packet based architecture is becoming an increasingly ubiquitous communications resource, providing the necessary underlying support for many services and applications. The classic voice call service over fixed circuit switched networks suffered a steep evolution with mobile networks and more recently another significant move is being witnessed towards packet based communications using the omnipresent Internet Protocol (IP) (Zourzouvillys & Rescorla, 2010). It is known that, due to real time requirements, voice over IP (VoIP) needs tighter delivery guarantees from the networking infrastructure than data transmission. While such requirements put strong bounds on maximum end to end delay, there is some tolerance to errors and packet losses in VoIP services providing that a minimum quality level is experienced by the users. Therefore, voice signals delivered over IP based networks are likely to be affected by transmission errors and packet losses, leading to perceptually annoying communication impairments. Although it is not possible to fully recover the original voice signals from those received with errors and/or missing data, it is still possible to improve the quality delivered to users by using appropriate error concealment methods and controlling the Quality of Service (QoS) (Becvar et al., 2007).

This chapter is concerned with voice signal reconstruction methods and quality evaluation in VoIP communications. An overview of suitable solutions to conceal the impairment effects in order to improve the QoS and consequently the Quality of Experience (QoE) is presented in section 2. Among these, simple techniques based on either silence or waveform substitution and others that embed voice parameters of a packet in its predecessor are addressed. In addition, more sophisticated techniques which use diverse interleaving procedures at the packetization stage and/or perform voice synthesis at the receiver are also addressed. Section 3 provides a brief review of relevant algebra concepts in order to build an adequate basis to understand the fundamentals of the signal reconstruction techniques addressed in the remaining sections. Since signal reconstruction leads to linear interpolation problems defined as system of equations, the characterization of the corresponding system matrix is necessary because it provides relevant insight about the problem solution. In such

characterisation, it will be shown that eigenvalues, and particularly the spectral radius, have a fundamental role on problem conditioning. This is analysed in detail because existence of a solution for the interpolation problem and its accuracy both depend on the characterisation of the problem conditioning. Section 4 of this chapter describes in detail effective signal reconstruction techniques capable to cope with missing data in voice communication systems. Two linear interpolation signal reconstruction algorithms, suitable to be used in VoIP technology, are presented along with comparison between their main features and performance. The difference between maximum and minimum dimension problems, as well as the difference between iterative and direct computation for finding the problem solution are also addressed. One of the interpolation algorithms is the discrete version of the Papoulis-Gerchberg algorithm, which is a maximum dimension iterative algorithm based on two linear operations: sampling and band limiting. A particular emphasis will be given to the iterative algorithms used to obtain a target accuracy subject to appropriate convergence conditions. The importance of the system matrix spectral radius is also explained including its dependence from the error pattern geometry. Evidence is provided to show why interleaved errors are less harmful than random or burst errors. The other interpolation algorithm presented in section 4 is a minimum dimension one which leads to a system matrix whose dimension depends on the number of sample errors. Therefore the system matrix dimension is lower than that of the Papoulis-Gerchberg algorithm. Besides an iterative computational variant, this type of problem allows direct matrix computation when it is well-conditioned. As a consequence, it demands less computational effort and thus reconstruction time is also smaller. In regard to the interleaved error geometry, it is shown that a judicious choice of conjugated interleaving and redundancy factors permits to place the reconstruction problem into a well conditioned operational point. By combining these issues with the possibility of having fixed pre-computed system matrices, real-time voice reconstruction is possible for a great deal of error patterns. Simulation results are also presented and discussed showing that the minimum dimension algorithm is faster than its maximum dimension counterpart, while achieving the same reconstruction quality. Finally section 6 presents a case study including experimental results from field testing with voice quality evaluation, recently carried out at the Research Labs of Portugal Telecom Inovação (PT Inovação). Based on these results, a Mean Opinion Score (MOS)-based quality model is derived from the parametric E-Model and validated using the algorithm defined by ITU-T Perceptual Evaluation of Speech Quality (ITU-T, 2001).

2. Voice signal reconstruction and quality evaluation

2.1 Voice signal reconstruction
Transmission errors in voice communications and particularly in voice over IP networks are known to have several different causes but the single effect of delivering poor quality of service to users of such services and applications. In general this is due to missing/lost samples in the signal delivered to the receiver.

Channel coding can be used to protect transmitted signals from packet loss but it introduces extra redundancy and still does not guarantee error-free delivery. In order to achieve higher quality in VoIP services with low delay, effective error concealment techniques must be used at the receiver. Typically such techniques extract features from the received signal and use them to recover the lost data.

The different approaches to deal with voice concealment can be classified in either source-coder independent or source-coder dependent (Wah et al., 2000). The former schemes implement loss concealment methods only at the receiver end. In such receiver-based reconstruction schemes, lost packets may be approximately recovered by using signal reconstruction algorithms. The latter schemes might be more effective but also more complex and in general higher transmission bandwidth is necessary. In such schemes, the sender first processes the input signals, extract the features of speech, and transmit them to the receiver along with the voice signal itself. For instance, in (Tosun & Kabal, 2005) the authors propose to use additional redundant information to ease concealment of lost packets.

Source-coder independent techniques are mostly based on signal reconstruction algorithms which use interpolation techniques combined with packetization schemes that help to recover the missing samples of the signal (Bhute & Shrawankar, 2008), (Jayant & Christensen, 1981).

Among several possible solutions, it is worth to mention those algorithms that try to reconstruct the missing segment of the signal from correctly received samples. For instance, waveform substitution is a method which replaces the missing part of the signal with samples of the same value as its past or future neighbours, while the pattern matching method builds a pattern from the last M known samples and searches over a window of size N the set of M samples which best matches the pattern (Goodman et al., 1986), (Tang, 1991). In (Aoki, 2004) the proposed reconstruction technique takes account of pitch variation between the previous and the next known signal frames.

In (Erdol et al., 1993) two reconstruction techniques are proposed based on slow-varying parameters of a voice signal: short-time energy and zero-crossing rate (or zerocrossing locations). The aim is to ensure amplitude and frequency continuity between the concealment waveform and the lost one. This can be implemented by storing parameters of packet k in packet k-1. Splitting the even and odd samples into different packets is another method which eases interpolation of the missing samples in case of packet loss. Particularly interesting to this work is an iterative reconstruction method proposed in (Ferreira, 1994a), which is the discrete version of the Papoulis Gerchberg interpolation algorithm.

A different approach, proposed in (Cheetham, 2006), is to provide mechanisms to ease signal error concealment by acting at packet level selective retransmissions to reduce the dependency on concealment techniques. Another packet level error concealment method base on time-scale modification capable of providing adaptive delay concealment is proposed in (Liu et al., 2001).

In practical receivers, the performance of voice reconstruction algorithms includes not only the signal quality obtained from reconstruction but also other parameters such as computational complexity which in turn has implications in the processing speed. Furthermore in handheld devices power consumption is also a critical factor to take into account in the implementation of these type of algorithms.

2.2 Voice quality evaluation methods

The Standardization Sector of International Telecommunication Union (ITU-T) has released a set of recommendations in regard to evaluation of telephony voice quality. These methods take into account the most significant human voice and audition characteristics along with possible impairments introduced by current voice communication systems, such as noise, delay, distortion due to low bitrate codecs, transmission errors and packet losses. Quality

evaluation methods for voice can be classified into subjective, objective and parametric methods. In the first case there must be people involved in the evaluation process to listen to a set of voice samples and provide their opinion, according to some predefined scale which corresponds to a numerical score. The Mean Opinion Score (MOS) collected from all listeners is then used as the quality metric of the subjective evaluation. The evaluation methods are further classified as reference and non-reference methods, depending on whether a reference signal is used for comparison with the one under evaluation. When the MOS scores refer to the listening quality, this is usually referred to as MOS_{LQS}[1] (ITU-T, 2006). If the MOS scores are obtained in a conversational environment, where delays play an important role in the achieved intelligibility, then this is referred to as MOS_{CQS}[2]. Even though a significant number of participants should be used in subjective tests (ITU-T, 1996), every time a particular set of tests is repeated does not necessarily lead to exactly the same results. Subjective testing is expensive, time-consuming and obviously not adequate to real-time quality monitoring. Therefore, objective tests without human intervention, are the best solutions to overcome the constraints of the subjective ones (Falk & Chan, 2009). Nowadays, the Perceptual Evaluation of Speech Quality (PESQ), defined in Rec. ITU-T P.862 (ITU-T, 2001), is widely accepted as a reference objective method to compute approximate MOS scores with good accuracy. Among the voice codecs of interest to VoIP, there are the ITU-T G.711, G.729 and G.723.1. Since the reference methods interfere with the normal operation of the communication system, they are usually known as intrusive methods.

The PESQ method transforms both the original and the degraded signal into an intermediate representation which is analogous to the psychophysical representation of audio signals in the human auditory system. Such representation takes into account the perceptual frequency (Bark) and loudness (Sone). Then, in the Bark domain, some perceptive operations are performed taking into account loudness densities, from which the disturbances are calculated. Based on these disturbances, the PESQ MOS is derived. This is commonly called the raw MOS since the respective values range from -1 to 4.5. It is often necessary to map raw MOS into another scale in order to compare the results with MOS obtained from subjective methods. The ITU-T Rec. P.862.1 (ITU-T, 2003) provides such a mapping function, from which the so-called MOS_{LQO}[3] is obtained.

Another standards, such as the Single-ended Method for Objective Speech Quality Assessment in Narrow-band Telephony Applications described in Rec. ITU-T P.563 (ITU-T, 2004), do not require a reference signal to compare with the one under evaluation. They are also called single-ended or non-intrusive methods.

The E-Model, described in the Rec. ITU-T G.107, (ITU-T, 2005) is a parametric model. While signal based methods use perceptual features extracted from the speech signal to estimate quality, the parametric E-Model uses a set of parameters that characterize the communication chain such as codecs, packet loss pattern, loss rate, delay and loudness. Then the impairment factors are computed to estimate speech quality. This model assumes that the transmission voice impairments can be transformed into psychological impairment factors in an additive psychological scale. The evaluation score of such process is defined by a rating factor *R* given by

---

[1] "Listening Quality Subjective"

[2] "Conversational Quality Subjective"

[3] "Listen Quality Objective"

$$R = R_0 - I_s - I_d - I_{e-eff} + A \qquad (1)$$

where R_0 is a base factor representative of the signal-to-noise ratio, including noise sources such as circuit noise and room noise, I_s is a combination of all impairments which occur more or less simultaneously with the signal transmission, I_d includes the impairments due to delay, I_{e-eff} represents impairments caused by equipment (e.g., codec impairments at different packet loss scenarios) and A is an advantage factor that allows for compensation of impairment factors. Based on the value of R, which is comprised between 0 and 100, Rec. ITU-T G.109 (ITU-T, 1999) defines five categories of speech transmission quality, in which 0 corresponds to the worst quality and 100 corresponds to the best quality. Annex B of Rec. ITU-T G.107 includes the expressions to map R ratings to MOS scores which provide an estimation of the conversational quality usually referred to as MOS_{CQE}[4]. If delay impairments are not considered, the I_d factor is not taken into account, and by means of ITU-T G.107 Annex B expressions, MOS_{CQE} is referred to as MOS_{LQE}[5].

3. Algebraic fundamentals

This section presents the most relevant concepts of linear algebra in regard to the voice reconstruction methods described in detail in the next sections. The most important mathematical definitions and relationships are explained with particular emphasis on those with applications in signal reconstruction problems.

Let us define $\mathbb{C}$, $\mathbb{R}$ and $\mathbb{Z}$ as the sets of complex, real and integer numbers respectively, and $\mathbb{C}^N$, $\mathbb{R}^N$ and $\mathbb{Z}^N$ as complex, real and integer N dimensional spaces. An element of any of these sets is called a vector. Let us consider f a continuous function. An indexed sequence $x[n]$ given by

$$x[n] = f(nT), \qquad n \in Z, \quad T \in R \qquad (2)$$

is defined as a sampled version of f.

A complex sequence of length N is represented by the column vector $x \in \mathbb{C}^N$ with components $[x_0, x_1, \ldots, x_{N-1}]^T$, where x^T is the transpose of x. In digital signal processing, such vector components are known as signal samples.

The solution of many signal processing problems is often found by solving a set of linear equations, i.e., a system of n equations and n variables $x_1, x_2, \ldots x_n$ defined as,

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2 \\ \vdots \\ a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n = b_n \end{cases} \qquad (3)$$

where elements $a_{ij}, b_i \in \mathbb{R}$. The above equation can be written in either matricial form,

---

[4] "Conversational Quality Estimated"

[5] "Listenen Quality Estimated"

$$
\begin{bmatrix}
a_{11} & a_{12} & \cdots & a_{n4} \\
a_{21} & a_{22} & \cdots & a_{2n} \\
\cdots & \cdots & \ddots & \cdots \\
a_{n1} & a_{n2} & \cdots & a_{nn}
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2 \\
\vdots \\
x_n
\end{bmatrix}
=
\begin{bmatrix}
b_1 \\
b_2 \\
\vdots \\
b_n
\end{bmatrix}
\tag{4}
$$

or in compact algebraic form

$$Ax = b \tag{5}$$

A is known as the system matrix and if $A=A^T$, then it is called a symmetric matrix. Let the complex number $\bar{z} = a - bi$ be the conjugate of $z = a + bi$, where i is the imaginary unit. The conjugate transpose of the $m{\times}n$ matrix A is the $n{\times}m$ matrix A^H obtained from A by taking the transpose and the complex conjugate of each element a_{ij}. For real matrices $A^H=A^T$ and A is normal if $A^TA=AA^T$. Any matrix A, either real or complex, is said to be hermitian if $A^H=A$. Denoting by I_n an $n{\times}n$ identity matrix, any $n{\times}n$ square matrix A is invertible or non-singular when there is a matrix B that satisfies the condition $AB=BA=I_n$. Matrix B is called the inverse of A, and it is denoted by A^{-1}. If A is invertible, then $A^{-1}Ax=A^{-1}b$ and the system equation $Ax=b$ has an unique solution given by

$$x = A^{-1}b \tag{6}$$

An $n{\times}n$ complex matrix A that satisfies the condition $A^HA=AA^H=I_n$, (or $A^{-1}=A^H$) is called an unitary matrix.

Considering an $n{\times}m$ matrix A and the index sets $\alpha=\{i_1, i_2, \dots i_p\}$ and $\beta=\{j_1, j_2, \dots, j_q\}$, with $p<n$ and $q<m$, a submatrix of A, denoted by $A(\alpha,\beta)$, is obtained by taking those rows and columns of A that are indexed by α and β, respectively. For example

$$
\begin{bmatrix}
1 & 2 & 3 \\
4 & 5 & 6 \\
7 & 8 & 9
\end{bmatrix}
(\{1,3\},\{1,2,3\}) =
\begin{bmatrix}
1 & 2 & 3 \\
7 & 8 & 9
\end{bmatrix}
\tag{7}
$$

If $\alpha=\beta$, the resulting submatrix is called a principal submatrix of A.

An eigenvector v of a square matrix A is a non-zero column vector that satisfies the following condition:

$$Av=\lambda v \tag{8}$$

for a scalar λ, which is said to be an eigenvalue of A corresponding to the eigenvector v. In other words, when A is multiplied by v, the result is the same as a scalar λ multiplied by v. Note that it is much easier to multiply a scalar by a vector than a matrix by a vector.

The spectrum of A is defined as the set of its eigenvalues, while the spectral radius of A, denoted by $\rho(A)$, is the supremum[6] among the absolute values of its spectrum elements. Since the number of eigenvalues is finite, the supremum can be replaced with the maximum. That is

$$\rho(A) = \max_i |\lambda_i| \tag{9}$$

---

[6] The supremum of a set S, sup$\{S\}$, is v if and only if: i) v is an upper bound for S and ii) no real number smaller than v is an upper bound for S (Kincaid & Cheney, 2002).

If $\rho(A)<1$, then the inverse of I-A exists and the system of (5) has a possible solution. This solution can be obtained by a direct calculation method as given in (6) or by an iterative method.

A vector norm can be thought of as the length or magnitude of vector x. Several types of norms are defined (Kincaid & Cheney, 2002). The most familiar norm is the Euclidian l_2-norm, defined as

$$\|x\|_2 = \left(\sum_{i=1}^{N} x_i^2 \right)^{1/2}$$

Other norms, such as the l_∞- and l_1-norms are also relevant,

$$\|x\|_\infty = \max_{1 \le i \le N} |x_i| \ , \ \|x\|_1 = \sum_{i=1}^{N} |x_i|$$

The matrix norm subordinate to a vector norm is defined as

$$\|A\| = \sup \left\{ \|Au\| : u \in \Re^N, \|u\| = 1 \right\}$$

Conditioning of a problem is another important concept, informally used to indicate how sensitive the solution of a problem is to small changes in the input data. A problem is said to be ill-conditioned if small changes in the input data produce large variations in the solution, whereas the solution of a well conditioned problem is less sensitive to variations in the input data.

For certain types of problems, a condition number can be defined as follows. Concerning the problem defined in (5), a perturbation on b will produce a corresponding perturbation on x, thus (5) can be written as

$$A\tilde{x} = \tilde{b} \tag{10}$$

where $\tilde{x}$ stands for the perturbation on x caused by the perturbation $\tilde{b}$ on b. The relation between relative perturbations is given by

$$\frac{\|x - \tilde{x}\|}{x} \le \|A\| . \|A^{-1}\| \frac{\|b - \tilde{b}\|}{b} \tag{11}$$

which permits to define the condition number as expression (Kincaid & Cheney, 2002)

$$k(A) = \|A\| . \|A^{-1}\| \tag{12}$$

Thus, if the condition number is large, even a small error in b may cause a large error in x. If the condition number is small, then the error in x will not be much higher than the error in b. The condition number is a property of the problem obtained from matrix A, which leads to well-conditioned problems whenever its value is close to unity. In the case where A is a normal matrix, the condition number assumes the form

$$k(A) = \left| \frac{\lambda_{\max}(A)}{\lambda_{\min}(A)} \right| \tag{13}$$

The eigenvalues of the system matrix and the relation between them play an important role in the problem conditioning. In this context, special attention should be paid to the spectral radius. As it will be explained in section 5, a spectral radius near or greater than 1 leads to a ill-conditioned problem whereas a smaller spectral radius between 0 and 1 leads to a well-conditioned problem.

Another important property is idempotence by which an operation can be repeated over the same data without changing the result. In algebra context, an nxn matrix A is said to be idempotent if $A^2=A$.

In a Toeplitz matrix A, each of its elements satisfies $a_{ij}=a_{i-j}$, which is equivalent to $a_{ij}=a_{i-1,j-1}$, thus, each descending diagonal from left to right is constant, as shown below.

$$A = \begin{bmatrix} a_0 & a_{-1} & a_{-2} & \cdots & a_{-(N-1)} \\ a_1 & a_0 & a_{-1} & \cdots & a_{-(N-2)} \\ a_2 & a_1 & a_0 & \ddots & \\ \vdots & \vdots & \ddots & \ddots & a_{-1} \\ a_{(N-1)} & a_{(N-2)} & \cdots & a_1 & a_0 \end{bmatrix}.$$

In the case of a complex matrix where a_{-k} is the conjugate of a_k, then it is called an hermitian Toeplitz whereas if the matrix is real, then it is a symmetric Toeplitz. In the system equation (5), if A is a mxn Toeplitz matrix, then the system has only $m+n-1$ degrees of freedom, rather than mxn.

A matrix A is positive definite if the associated quadratic form is positive, i.e., if $x^H A x > 0$, $\forall x \neq 0$. If A is positive definite and symmetric, then all of its eigenvalues λ_i are real and positive (Kincaid & Cheney, 2002). Every positive definite matrix is invertible and its inverse is also positive definite (Horn & Johnson, 1985). A matrix A is non-negative definite if the associated quadratic form is non-negative, that is, if $x^H A x \geq 0$, $\forall x \neq 0$.

There are several possible methods to find the solution of (5), which may be classified in either direct or iterative methods. Concerning the direct methods, the solution can theoretically be found by left-multiplying by A^{-1}, if it is known, resulting in the equation (6). There are several approaches, from Gauss-Jordan elimination to factorization methods such as LU decomposition. Some special structures of A can lead to simple solutions. As an example, equation (5) has a trivial solution when matrix A is diagonal. In this case, the solution is

$$x = \begin{bmatrix} b_1 / a_{11} \\ b_2 / a_{22} \\ \vdots \\ b_n / a_{nn} \end{bmatrix} \tag{14}$$

If $a_{ii}=0$ and $b_i=0$, for any i, then x_i may be any real number. If $a_{ii}=0$ and $b_i\neq0$ there is no solution for the system. If the entries below or above the main diagonal of a mxn matrix A are zero, then A is either a lower (L) or upper (U) triangular matrix, respectively, as follows.

$$L = \begin{bmatrix} l_{11} & 0 & \cdots & 0 \\ l_{21} & l_{22} & \cdots & 0 \\ \cdots & \cdots & \ddots & \cdots \\ l_{n1} & l_{n2} & \cdots & l_{nn} \end{bmatrix} \qquad U = \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ 0 & u_{22} & \cdots & u_{2n} \\ \cdots & \cdots & \ddots & \cdots \\ 0 & 0 & \cdots & u_{nn} \end{bmatrix}$$

If all entries of the main diagonal of a triangular matrix are zero, then such matrix is called either strictly upper or strictly lower triangular. Assuming a lower triangular matrix A and $a_{ii} \neq 0$, $\forall i$, equations from (5) become

$$\begin{bmatrix} a_{11} & 0 & \cdots & 0 \\ a_{21} & a_{22} & \cdots & 0 \\ \cdots & \cdots & \ddots & \cdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

and then obtaining x_1 from the first equation becomes trivial.

LU decomposition is a simplified method to solve a system of equations where matrix A can be defined as the product between a lower triangular matrix and an upper triangular matrix, i.e., $A=LU$. Thus, the linear set of equations to solve becomes $Ax=(LU)x=L(Ux)=b$ and the solution can be found by first solving it for vector y such that $Ly=b$ and then solving for x, using $Ux=y$. The advantage of breaking up one linear set of equations into two successive ones is that the solution of a triangular set of equations is quite trivial (Press et al., 2007).

A particular case of LU decomposition is the Cholesky decomposition where decomposition of matrix A is given by the product of a lower triangular matrix with its conjugate transpose, i.e., $A=LL^T$ and L is a lower triangular matrix with all diagonal elements positive. When applicable, the Cholesky decomposition is about twice as fast as other methods used for solving systems of linear equations (Press et al., 1994). Note that this method requires that A is real, symmetric and positive-definite.

Direct methods to resolve the type of equations such as (6) ideally produce a solution correct to machine accuracy. However, when the system order is high, with thousands of equations, the computational effort may be critical either in terms of execution time or other resources like memory. In this case, iterative methods might be the answer to overcome such constraints. In their *modus operandi*, iterative methods produce a sequence of vectors that converge to the final solution as the computational process evolves. The process halts when either some pre-defined number of iterations is reached or an acceptable level of accuracy is obtained at any possible iteration. In high dimensional systems, if precision is not a strong requirement, it is possible to approximate the solution with just a few iterations. Particularly, in sparse systems, where the number of zero entries in the iteration matrix is high, iterative methods prove to be very efficient in the sense that only a small number of computations are necessary.

There are several specific iterative methods particularly suited to solve systems of the form of (5). Among them, Jacobi and Gauss-Seidel methods are paradigmatic. The Jacobi method follows from the individual analysis of each of the n system equations as defined in (5). If the following expression holds for the ith equation,

$$\sum_{j=1}^{n} a_{ij} x_j = b_i \tag{15}$$

then x_i can be solved assuming that other entries do not vary, i.e.,

$$x_i = \left(b_i - \sum_{j=1}^{n} a_{ij} x_j \right) \Big/ a_{ii}, \qquad j \neq i \tag{16}$$

which suggests an iterative resolution for the ith equation, as given by

$$x_i^{(k)} = \left(b_i - \sum_{j=1}^{n} a_{ij} x_j^{(k-1)}\right) \Big/ a_{ii}, \qquad j \neq i \qquad (17)$$

The Gauss-Seidel method can be seen as an enhancement of Jacobi method in which updated values of x_i on the right-hand side of (17) are used as soon as they become available in the same iteration. That is, instead of use x_i from iteration k-1 in iteration k, the value of x_i from previous equation is used in the same iteration when available. The first equation is the only exception. As an example, for the first two equations, we have

$$a_{11} x_1^{(1)} = b_1 - a_{12} x_2^0 - a_{13} x_3^{(0)} \cdots - a_{1n} x_n^{(0)}$$
$$a_{22} x_2^{(1)} = b_2 - a_{21} x_1^{(1)} - a_{23} x_3^{(0)} \cdots - a_{1n} x_n^{(0)}$$
$$\vdots$$

which leads to

$$x_i^{(k)} = \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k-1)}\right) \Big/ a_{ii} \qquad (18)$$

Between iterations k-1 and k, the result converges to the final solution an amount given by $\delta x^{(k)} = x^{(k)} - x^{(k-1)}$. Thus, the solution at iteration k is given by $x^{(k)} = x^{(k-1)} + \delta x^{(k)}$. However, at iteration k-1 the result $x^{(k-1)}$ differs from the final solution by the amount of $\Delta x^{(k)} = x - x^{(k-1)}$, which is different from $\delta x^{(k)}$. Since $\delta x^{(k)} < \Delta x^{(k)}$, one may speed up the convergence rate by using the over-relaxation form (19) instead of simply computing $x^{(k)}$ as $x^{(k)} = x^{(k-1)} + \delta x^{(k)}$.

$$x^{(k)} = x^{(k-1)} + \omega x^{(k)}, \qquad \omega > 1 \qquad (19)$$

where w is called the relaxation factor. Typically, this value is constant for all k and $1 < w < 2$. The iterative process expressed in (19) is called Successive Over-Relaxation, commonly abbreviated as SOR.

The basic concepts of linear algebra presented above are used in problems of voice signal reconstruction dealing with missing samples, such as those described in this chapter. These problems can be defined as linear system equations in which interpolation algorithms play an important role in finding their solutions. In this context, missing or unknown samples due to transmission errors or data loss are set to zero at the receiver, which in turn shall use reconstruction methods to find the best possible estimate of the original signal.

4. Two linear interpolation algorithms

This section describes two reconstruction algorithms capable of computing accurate estimates of missing samples in voice signals due to packet loss, transmission errors, etc. Therefore these algorithms are suitable to be implemented in receivers as error concealment methods to enhance the QoE delivered to users. Both algorithms are based on linear interpolation and operate on a sequence of voice samples of a predefined length, i.e., the number of samples under processing at a given time is constant. Let us define a N-dimension signal vector with Fourier components $x_1, x_2, \ldots x_N$, and the Fourier matrix F a unitary NxN matrix with components F_{mk} given by

$$F_{mk} = \frac{1}{\sqrt{N}} e^{-i\frac{2\pi}{N}mk} \qquad\qquad (20)$$

where i is the imaginary unit. Therefore, the Discrete Fourier Transform DFT of x, here represented by $\hat{x}$, is the sequence $\hat{x} = Fx$. In this context, sampling and band-limiting are two relevant linear operations defined in $\mathbb{C}^N$ which should be recalled. In this case, sampling is defined as a mapping function which converts a sequence of samples (i.e., a digital signal), into another one by setting to zero some of the original samples. This is used as an error modelling function by which the sampled version of the signal (i.e., the one with missing/lost samples) may be obtained by multiplying the original signal with a diagonal matrix D whose elements are comprised of zeros and ones (Ferreira, 1994a). The resulting signal is called the observed signal and it corresponds to a corrupted version of the original one. Therefore D is called the sampling matrix and its diagonal is the sampling set associated with the sampling operation. Considering s the number of nonzero entries in the sampling set, then s/N defines the density of sampling. Here it is assumed that $s<N$ and D is not the identity matrix, I.

Band-limiting can also be viewed as a sampling operation, in which the signal samples set to zero are in the Fourier domain, i.e., signal frequency components. In fact, by multiplying a diagonal matrix Γ by Fx, the resulting matrix ΓFx has zeros in those spectral components of x that correspond to the zeros of Γ. Then by left-multiplying F^{-1} by ΓFx returns the signal into the time domain, resulting in a filtering operation. Therefore, such band-limiting operation can be defined by a linear operator characterized by a matrix B defined as $B=F^{-1}\Gamma F$. As mentioned above, Γ is a sampling matrix different from the identity I. The bandwidth of the signal $y=Bx$ is defined as q/N, where q is the number of nonzero entries in Γ. The Nyquist sampling frequency is denoted as f_s while f_{os} is an oversampling frequency. In this case, an oversampling factor r is defined as $r=f_s/f_{os}$. Such oversampling factor is also given by $r=q/N$ and if $r<1$ then there is redundancy in the signal.

Considering N samples of a voice signal and n the number of corrupted samples, then the following condition holds: $n<N$. If the reconstruction algorithm has to solve N equations, i.e., using the whole space of dimension N, then it is called a maximum dimension algorithm. However, if the algorithm only needs to solve n equations concerning just the unknown samples, then it is called a minimum dimension algorithm. The error geometry is defined as the pattern of missing samples within the whole sequence of samples. Depending on the relative position between missing samples, three geometries are addressed: i) interleaved geometry, where the missing samples are equidistant and multiple of an integer $l \geq 2$; ii) burst geometry where the missing samples occur in bursts of contiguous samples and iii) random geometry where the missing samples do not exhibit any special pattern but are randomly distributed along the original sequence. In this context a signal with bandwidth b means that the highest normalized frequency in the signal is $b/2$. The nonzero entries of the Γ diagonal define the so-called passband of B (Ferreira, 1994a).

Moreover, note that both sampling and band-limiting are idempotent operations. This means that repeating such operations over the same signal always produce the same result as that obtained from one single operation. Therefore, idempotence allows defining a passband signal x as follows:

$$x=Bx. \qquad\qquad (21)$$

4.1 A maximum dimension algorithm – the discrete version of Papoulis-Gerchberg

The discrete version of Papoulis-Gerchberg algorithm is an iterative linear interpolation algorithm (Ferreira, 1994a). Its aim is to recover missing samples in a finite-length, band-limited data sequence x, given their positions within the sequence. In this case the data sequence of interest is a time segment of a voice signal. Fig. 1 and Fig. 2 show an example of both original and observed signals, x and y respectively, where the last one is obtained by setting to zero two of the original samples.
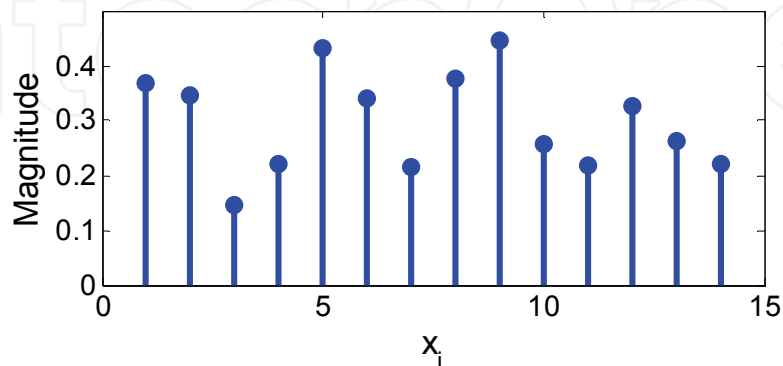
Fig. 1. Original time-domain signal x
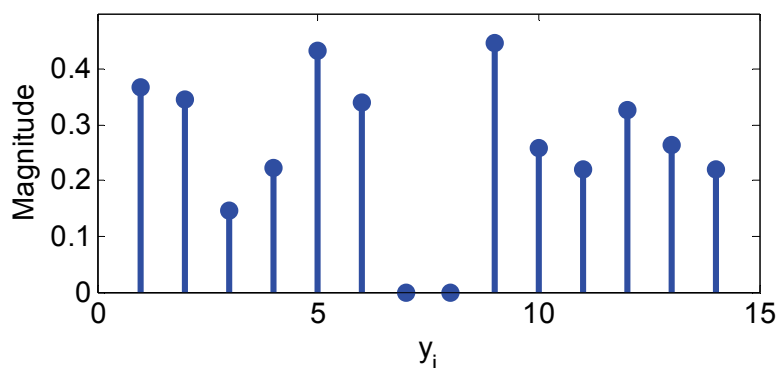
Fig. 2. Observed time-domain signal y

In this signal reconstruction algorithm, the known data are the samples of the observed signal, the position of the missing ones and the bandwidth of the original signal, as given by Equation (21). This is equivalent to know the vector y (Fig. 2), and the matrices D and B referred to above. Note that, in practice matrix D is obtained from the received signal with lost samples.
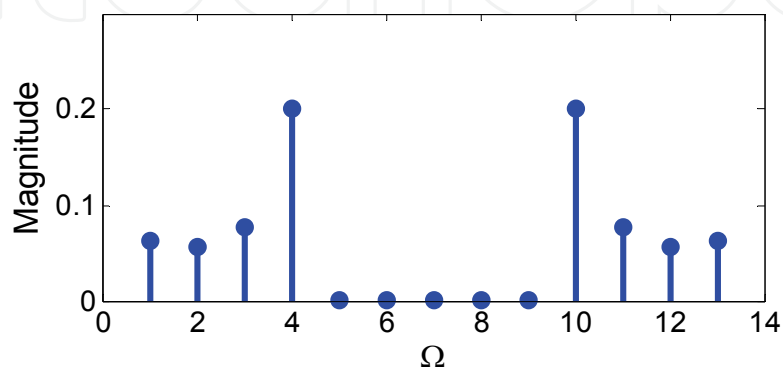
Fig. 3. Spectral components of the original signal, x

The aim of the reconstruction algorithm is to make the observed signal as close as possible to the original one. By knowing the original signal bandwidth, from which the spectral components are derived, it is possible to compare the observed signal with the original one as the iterative process converges. Fig. 3 shows the spectral components of the original signal x without the DC component.

Herein, the main algorithmic steps leading to the reconstructed signal are described as follows.

Step 1 – Compute the DFT of y: $DFT(y)=Fy$

The first step in this algorithm is to transform y into the frequency domain, by computing its DFT, i.e., $DFT(y)=Fy$. In the subsequent iteration process, the observed signal y is subject to several operations. Let us define $y^{(0)}$ as iteration 0 of the reconstructed signal, $y^{(1)}$ the result of the first iteration, etc. Iteration 0 is obtained as $y^{(0)}=y=Dx$. As expected, whenever a signal incurs in sharp time-domain variations, such as those originated by loss of samples, this implies changes in the frequency domain. Therefore when losses occur in the original signal, high frequency components appear in the observed signal y, which lie outside the bandwidth of the original signal. Fig. 4 shows the result of such operation, where high frequency components (i.e., central components) appeared at locations where originally there were zeros (see Fig. 3).
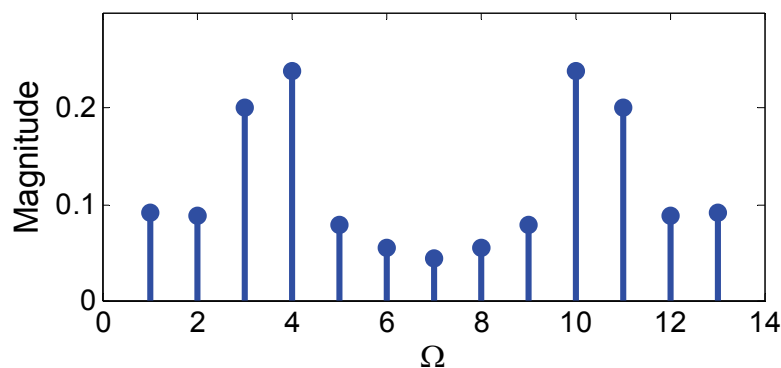
Fig. 4. Spectral components of the observed signal, y

Step 2 – Filter y according to the spectral characteristic of x: $DFT(y')=\Gamma Fy$

The underlying idea behind this process of signal reconstruction is to filter the observed signal y with the same spectral characteristics as those of the original signal. Then, the filtered signal, y', is closer to x than y, because its transform domain representation was also approximated to the original one.

Such filtering operation is achieved by left-multiplying matrix Γ by $DFT(y)$. It is given by $DFT(y')=\Gamma\ Fy$. Fig. 5 shows the result of this filtering operation, where the undesirable spectral components become zero while the others remain unchanged.

Step 3 – Return to the time-domain: $y'=F^{-1}\Gamma Fy$

The filtered signal y' can now be obtained in the time domain through the inverse of $DFT(y')$. Note that, as pointed out above, y' is closer to the original signal x than y, i.e., the effect of filtering is to approximate the missing samples towards their original values. Fig. 6 shows these new samples growing at the sampling instants where their previous values were zero. After few initial iterations, their amplitudes are not yet exactly the same as the original ones, but they tend to the original ones as the iterative process converges to a more accurate solution. This is because the values of such samples result from a filtering operation
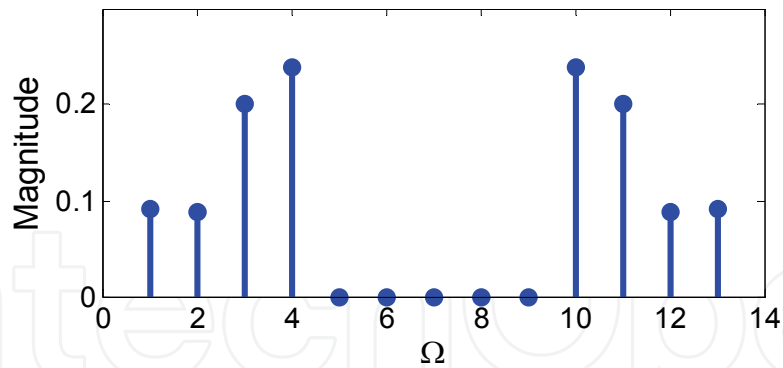
Fig. 5. Spectral components of y' after filtering y : $DFT(y')$
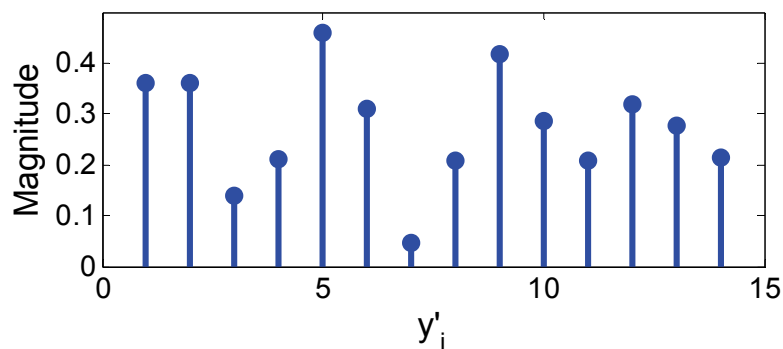

Fig. 6. Signal y' obtained after enforcing the original signal bandwidth through filtering.

that makes the corrupted signal closer to the original one by forcing it to have similar spectral components. However, the filtering operation also changes the values of the non-corrupted samples of y.

Step 4 – Extract the reconstructed samples from the others: $y''=(I-D)y'$.

Although the previous process has the advantage of recovering sample values at sampling instants where they were zeros, it also slightly corrupts the good samples of the observed signal y. However, since the time locations of the missing samples are known through matrix D, in each iteration is possible to extract only the reconstructed samples through the following operation $y''=(I-D)y'$. The output of such extraction process is illustrated in Fig. 7 where only the new samples are left and all others are set to zero.
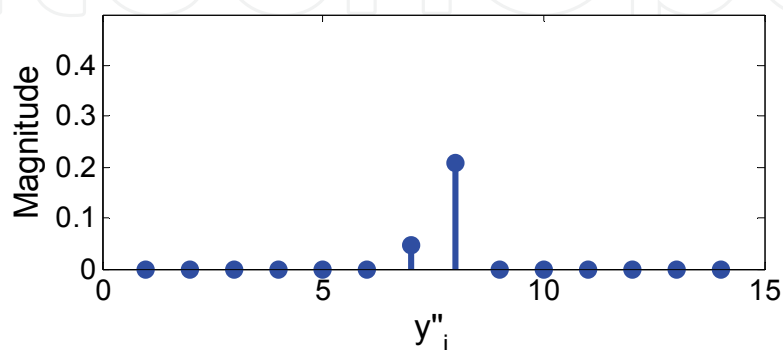
Fig. 7. Signal y'': samples extracted from y'.

Step 5 – Reconstructed signal composition: $y^{(1)}=y''+y$

After the previous steps, on the one hand, signal y'' has only non-zero samples at those sampling instants where the missing samples were located in the observed signal. On the other hand, at the remaining sampling instants, the observed signal y contains all non-corrupted samples. This means that signals y and y'' contain non-zero samples at mutually exclusive temporal instants. So the sum of both signals results in the first approximation $y^{(1)}$ of the original signal x and acomplishes the first iteration of the reconstruction process. Fig. 8 shows the reconstructed signal $y^{(1)}$ after the first iteration.
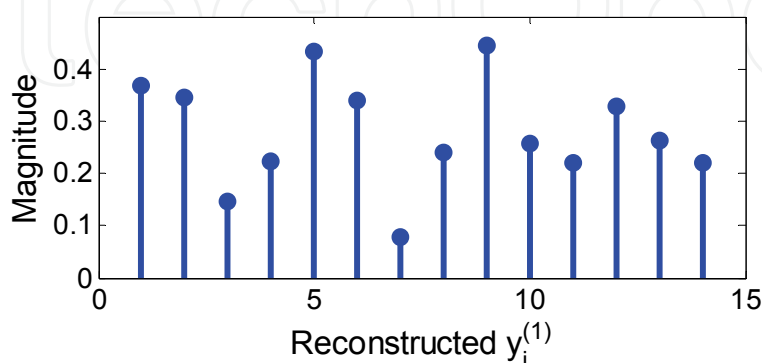
Fig. 8. Reconstructed signal $y^{(1)}$ after the first iteration.

Step 6 – Prepare next iteration: $y=y^{(1)}$.

For the next iteration, the reconstructed signal $y^{(1)}$ appears as the new observed signal to enter into a new reconstruction cycle. Therefore, at this step $y=y^{(1)}$. Note that, since the non-corrupted samples of the original observed signal are needed in Step 5 of each iteration, y must be stored in memory at the startup of the process, i.e., before Step 1.

The algorithmic steps described above can be defined by a sequence of algebraic expressions, which are the basis for software implementation of the reconstruction method. The following expressions fully describe the reconstruction algorithm from Step 6 to Step 1.

$$y^{(1)}=y'' + y \tag{22}$$

$$y^{(1)}=(I-D)y' + y \tag{23}$$

$$y^{(1)} = (I-D)F^{-1}\Gamma F y^{(0)} +y \tag{24}$$

From (24) it is possible to obtain the following expression for the reconstructed signal in the next iteration

$$y^{(k+1)} = (I-D)By^{(k)} + y \tag{25}$$

where $y^{(k)}$ represents the reconstructed signal at iteration k.

The iteration matrix is, thus, given by

$$A=(I-D)B \tag{26}$$

In order to improve convergence a relaxation constant, μ is used, thus Equation (25) becomes,

$$y^{(k+1)} = (I-\mu D)By^{(k)} + \mu y \tag{27}$$

and the iteration matrix is given by

$$A_\mu = (I - \mu D)B \tag{28}$$

For an effective use of such a constant, values of μ belonging to the interval $]0, 2[$ must be used (Ferreira, 1994a). More precisely, the optimum value of μ is given by

$$\mu_{opt} = \frac{2}{2 - \lambda_{\max}}$$

where $\lambda_{max} = \rho(S_1)$ with S_1 being $S_1 = B(I-D)B$.

The algorithm converges if the density of sampling is greater than the signal bandwidth, i.e., $s/N > q/N$. Thus, convergence can be guaranteed by reducing the number of missing samples and/or the bandwidth.

An important issue concerning convergence are the error patterns (i.e., location of missing samples) which influence the asymptotic convergence rate of the algorithm, for a given density. The convergence rate partially depends on matrix B and on the error geometry. In fact, for low pass signals, the best possible error patterns are those in which the missing sample time positions are equidistant (Neves et al., 2008). The worst possible geometry is that of contiguous missing samples. In the middle there is the random geometry. This issue is important in order to obtain a well-conditioned problem. The fact that this is a maximum dimension method, the possibility of low convergence rates and the computing resources required per iteration (essentially a pair of FFTs) are disadvantages of this approach. Since this is a maximum dimension problem it is expected to exhibit a relatively low convergence rate due to the enormous computing resources required. This is further discussed in section 5.

4.2 A minimum dimension algorithm

As mentioned in previous sections, a minimum dimension algorithm is characterised by a system of only n equations corresponding to the n unknown samples. This subsection describes a minimum dimension algorithm which also requires band-limited signals of finite-dimension similarly to the Papoulis-Gerchberg algorithm described above, i.e., equation (21) must be valid.

To establish the basic concepts of this algorithm, the specific case of an original signal x_i with length $N=5$ is used, i.e., $x_i = \{x_1, x_2, x_3, x_4, x_5\}$. For this signal, Equation (21) becomes

$$\begin{aligned} x_1 &= b_{11}x_1 + b_{12}x_2 + b_{13}x_3 + b_{14}x_4 + b_{15}x_5 \\ &\vdots \\ x_5 &= b_{51}x_1 + b_{52}x_2 + b_{53}x_3 + b_{54}x_4 + b_{55}x_5 \end{aligned} \tag{29}$$

where b_{ij} are the elements of the matrix B.

For reconstruction purposes let us assume that the 2nd and 4th samples of x_i are lost. Then the set of equations (29) are limited to those including the lost samples. In each of these equations, we are interested in separating the right side terms containing unknown samples (x_2, x_4) from those containing the known ones. This yields,

$$\begin{aligned} x_2 &= b_{21}x_1 + b_{22}x_2 + b_{23}x_3 + b_{24}x_4 + b_{25}x_5 \\ x_4 &= b_{41}x_1 + b_{42}x_2 + b_{43}x_3 + b_{44}x_4 + b_{45}x_5 \end{aligned} \tag{30}$$

which is equivalent to

$$\begin{bmatrix} x_2 & x_4 \end{bmatrix} = \begin{bmatrix} b_{22} & b_{24} \\ b_{42} & b_{44} \end{bmatrix}\begin{bmatrix} x_2 \\ x_4 \end{bmatrix} + \begin{bmatrix} b_{21} & b_{23} & b_{25} \\ b_{41} & b_{43} & b_{45} \end{bmatrix}\begin{bmatrix} x_1 \\ x_3 \\ x_5 \end{bmatrix} \tag{31}$$

Let us denote by u the subset of the original signal x_i which contains the unknown values. In this case, $u=\{x_2, x_4\}$ is of cardinality $k=2$. Also, let us define $U=\{i_1, i_2, \ldots, i_k\}$ as the set of subscripts of k unknown samples in x_i. In the present case, $U=\{2, 4\}$. Therefore, equations (31) can be written as

$$x_i = \sum_{j \in U} b_{ij}x_j + \sum_{j \notin U} b_{ij}x_j; \qquad i \in U \tag{32}$$

or, in matricial form

$$u = Su + h \tag{33}$$

where S is a kxk principal submatrix of B, as defined in (31), and h, is the $(N-k)$-dimensional vector in the second sum of (32), which is a linear combination of the known samples of x_i.
The conditions under which these equations provide a solution for u can be found in (Ferreira, 1994b). In the case where a noniterative method is used, equation (33), becomes equivalent to

$$\begin{aligned} u &= Su + h \\ u - Su &= h \\ Iu - Su &= h \\ (I-S)u &= h \\ (I-S)^{-1}(I-S)u &= (I-S)^{-1}h \\ u &= (I-S)^{-1}h \end{aligned} \tag{34}$$

This result is valid, providing that $(I-S)^{-1}$ exists. Thus, theoretically, Equation (33) has a unique solution regardless the number and distribution of the lost samples. If equation (33) is solved through an iterative process, then the following form is suggested in the case where a non-relaxation method is used.

$$u^{(i+1)} = Su^{(i)} + h \tag{35}$$

Then $u^{(k)}$ is obtained at iteration k and the solution is given by the limit

$$u = \lim_{i \to \infty} u^{(i)} \tag{36}$$

regardless of $u^{(0)}$. The condition $\rho(S)<1$ guarantees that such limit exists, where S is the system matrix (Ferreira, 1994b).
Two different techniques can be used to solve (33): Direct calculation and iterative methods. Direct calculation of u as given in (34) has the advantage of being done in one single step, providing that $(I-S)^{-1}$ exists. In practice, there are several factors which may lead to serious difficulties in calculating the inverse of $I-S$. For example, if one of the eigenvalues of S is close enough to unity, then computation of $(I-S)^{-1}$ may become very difficult, or even impossible, leading to an ill-conditioned problem. In such cases, an iterative method may be

used to circumvent this difficulty and to find an accurate approximation for solution u. Despite the fact of having an ill-conditioned problem, in the case of direct calculation such problem is impossible to solve whereas in the case of iterative methods an approximation is always possible to be found, though its accuracy may not be very high.

The eigenvalues of the system matrix S depend on the distribution of the missing samples. In particular, its spectral radius is more likely be unitary for burst distributions rather than for equidistant missing samples (Ferreira, 1994c). In the case of signal reconstruction it is interesting to note that, if the distribution of the missing samples $U=\{i_1, i_2, …, i_n\}$ is equidistant by some fixed integer $m \geq 1$, that is, $U=\{i_1m, i_2m, …, i_nm\}$, then the eigenvalues λ_i of S have an upper bound given by $(\lfloor rm \rfloor + 1)/m$ and a lower bound given by $\lfloor rm \rfloor / m$, i.e.,

$$\frac{\lfloor rm \rfloor}{m} \leq \lambda_i(S) \leq \frac{\lceil rm \rceil}{m} \leq 1 \tag{37}$$

where $\lfloor rm \rfloor$ denotes the greatest integer less than or equal to rm and $\lceil rm \rceil$ denotes the smallest integer equal or greater than rm. In the particular case of $r = \lfloor rm \rfloor / m$, the eigenvalues of S are all the same $\lambda_i(S) = r$, $\forall i$. In such case $S = rI$. In the particular case of $i_k = km$, the missing samples are equidistant and S becomes Toeplitz.

Given the above analysis, it is possible to put the problem into a well-conditioning point by properly selecting the gap between the missing samples. Then it is possible to put $\lambda_i(S)$ close to either r or its multiples, regardless of the number of missing samples. By using an appropriate choice of the oversampling and interleaving factors r and m (i.e., such that $m \times r$ is an integer) respectively, it is possible to put $\lambda_i(S)$ less enough than unity in order to control the reconstruction accuracy and processing speed.

In VoIP context, the use of an adequate interleaving factor m, not only makes such a signal to be more robust to possible degradations by transforming burst errors in equidistant ones, but also makes reconstruction easier because it leads to a well-conditioned problem. Therefore, when a packet is lost with n voice samples in its payload, this leads to a reconstruction problem where missing samples are equidistantly distributed, separated by m-1, and the matrix S of the resulting reconstruction problem is of dimension $n \times n$.

Fig. 9 shows the maximum and the minimum eigenvalues of S as a function of the interleaving factor, m, for a given bandwidth, defined by r. As the figure shows, greater values of m lead to well-conditioning problems because λ_{max} decreases as m increases. Also, when the product $r \times m$ is an integer, all eigenvalues are equal since they are $\lambda_i(S) = r$, as stated before. In Fig. 9, this occurs for m=5 and m=10.
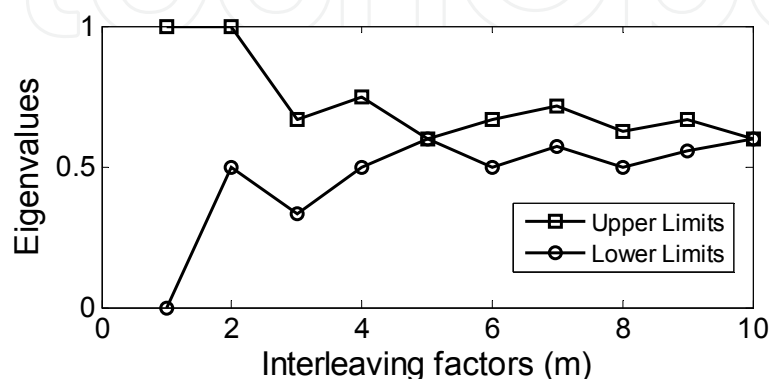

Fig. 9. Spectral radius vs. interleaving factor, r=0.6

Considering real time implementation issues, while the solution of (33) must be found in real time to be effective, the possibility to have a pool of different dimension system matrices *S*, previously calculated and stored in memory, turns the whole reconstruction process more expedite. Furthermore, its calculation may become as trivial as *S=rI*.

5. Simulation results

In this section, the reconstruction algorithms previously described are evaluated through simulation and the results are analysed and discussed. The simulation study is also aimed to provide a deeper understanding of the most important factors influencing the problem of signal reconstruction and to show how much a minimum dimension algorithm enhances the performance of the maximum dimension one.

In the case of the Papoulis-Gerchberg algorithm, it is important to analyse the factors that influence the conditioning of the reconstruction problem and how they influence its solution. These include the spectral radius of the iteration matrix, the distribution of the missing samples and the signal bandwidth. The performance is evaluated by measuring the number of iterations necessary to reach the solution, the percentage of lost samples, the iteration matrix spectral radius and the Root Mean-Squared Error (RMSE) between the original and reconstructed signals. The RMSE is given by the following expression, where $x[i]$ is the original signal, $\tilde{x}[i]$ the reconstructed signal and N the sequence length,

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(x[i]-\tilde{x}[i]\right)^2} \qquad (38)$$

The stop criterion was defined as an upper bound for residual error between consecutive iterations, given by

$$residual = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(y^{(k+1)}(i)-y^{(k)}(i)\right)^2} \leq 10^{-8} \qquad (39)$$

where $y^{(k)}(i)$ is the signal at iteration k. In the case of random geometry tests, i.e., those where the loss pattern of voice samples is random, each simulation run uses more missing samples than the previous one, which in turn acts as a seed in order to guarantee an increasing spectral radius over successive runs. The same voice signal was used in all the tests. In all experiments, a voice signal with $N=256$ samples was used. In the case of the Papoulis-Gerchberg algorithm, three different error distributions, referred to as interleaved, random and burst geometries were used. These experiments run on a computer equipped with an Intel T2300@1.66 MHz processor and 1.5 GB RAM. Also, two different signal bandwidths were used, defined by two different factors *r*.

5.1 The maximum-dimension Papoulis-Gerchberg algorithm
The performance of Papoulis-Gerchberg algorithm was evaluated by carrying out three tests intended to find out how the spectral radius of the iteration matrix, the error geometry and the signal bandwidth influence the convergence of the algorithm.

5.1.1 The spectral radius of the iteration matrix
This test is intended to evaluate the influence of the spectral radius of the iteration matrix in the algorithm convergence. In the experiments the oversampling factor was set to *r*=0.6 and

the relaxation constant μ=1. The percentage of missing samples varied from 0.4% to 25%. Fig. 10 shows the number of iterations necessary to obtain a residual error less than 10⁻⁸, as a function of the spectral radius. As one can observe, as the spectral radius of the iteration matrix increases, the number of iterations also increases following approximately an exponential function. The absolute error of the approximated solution obtained after iteration k+1 is bounded by

$$\|e_{k+1}\| \le \lambda_{\max}^k \|e_1\| \tag{40}$$

where λ_{max} represents the maximum eigenvalue of the system matrix (Ferreira, 1994a). This expression shows that in order to attain a given error, higher values of λ_{max} imply higher number of iterations, since λ_{max} is less than 1, k is greater than 1 and $\|e_1\|$ is constant. It is also possible to see that, if λ_{max}=1, the error after iteration k+1 will never decrease below to that of the first iteration, thus the algorithm does not converge. In the figure, it is evident that $\rho(A)$=1 leads to a non-convergence situation. Therefore, an important conclusion is that a spectral radius near to 1 can easily turn the problem into ill-conditioned making convergence difficult or even impossible. In this case an acceptable solution cannot be found.
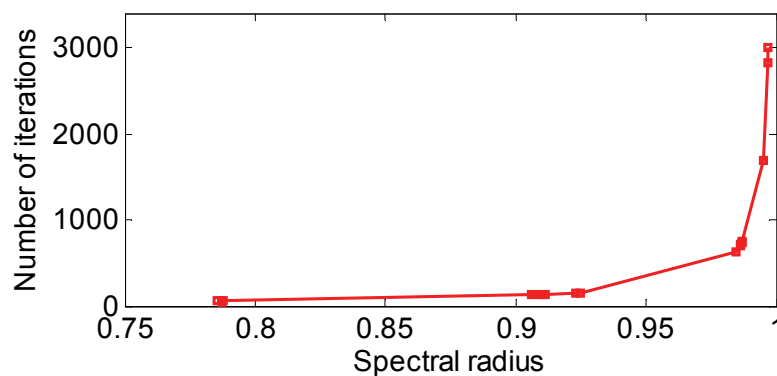
Fig. 10. Number of iterations *versus* the spectral radius (r=0.6)

5.1.2 Error geometry
This test is intended to evaluate the influence of the missing samples distribution on convergence and to identify the break even points, i.e., the maximum percentage of missing samples for which the algorithm is convergent. Note that each break even point also corresponds to a specific spectral radius because this is implicitly defined by the missing samples. In the experiments the values r=0.8 and μ=1 were used. Interleaved, random and burst distributions with loss percentages ranging from 1% to 50% were used. To determine break even points, spectral radii close to 1 were used.
Fig. 11 shows the spectral radius as a function of the percentage of missing samples, for the three error geometries under study. As the figure shows, the spectral radius depends on two factors: the percentage of missing samples and the error geometry. In regard to the percentage of missing samples, one can observe that the spectral radius increases as more samples are missing in the signal. This is common to all three geometries, which exhibit the same behaviour.
In the case of different error geometries, one can also see that for a given spectral radius, the interleaved geometry is the one which tolerates more missing samples and the burst

geometry is the one that tolerates less missing samples, since just two or three missing samples make the spectral radius close to 1. The behaviour of the random geometry is between the other two. From another point of view, for the same number of missing samples, the interleaved geometry exhibits a lower spectral radius than the others and the burst geometry exhibits the greatest spectral radius. These results lead to the conclusion that interleaved geometry is the error pattern that tolerates a greater percentage of missing samples in the signal.
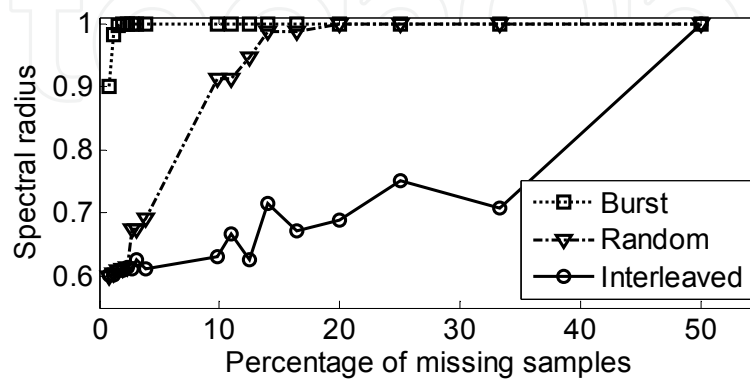
Fig. 11. Spectral radius *versus* percentage of missing samples, for three error geometries

Fig. 12 shows the break even points, for each type of error geometry defined by either the percentage of missing samples or its corresponding spectral radius. Note that these are the maximum values for which the algorithm still converges to a unique solution.
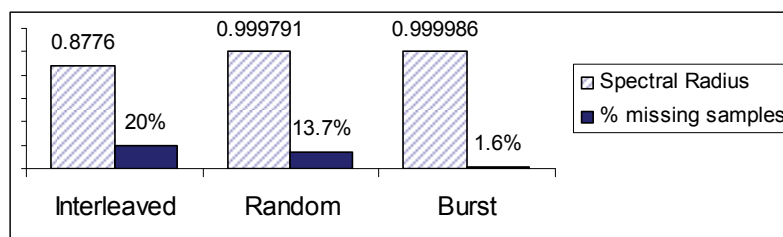
Fig. 12. Break even points for each geometry (r=0.8)

These results show that break even point positions vary according to the error geometry. In the case of the interleaved geometry, the maximum spectral radius that still leads to a well-conditioned problem is 0.8776. It corresponds to an interleaving factor of missing samples of m=5 (i.e., 1 out of 5) thus to 20% of missing samples. Other spectral radii between 0.8776 and 1 are possible, but the next value for the interleaving factor is four (m=4), which leads to a spectral radius of 1, thus to a ill-conditioned problem.

In the case of the random geometry, the maximum spectral radius that still leads to a well-conditioned problem is 0.999791 corresponding to lose 13.7% of the signal samples. Finally, the worst situation is the burst geometry in which the maximum possible spectral radius is 0.999986 corresponding to lose 1.6% of the signal samples.

Overall, these results confirm that the interleaved error geometry is the most tolerant to losses in the sense that more signal samples may be lost before the problem becomes ill-conditioned. On the opposite side, the burst error geometry was found to be the less tolerant to losses.

5.1.3 Influence of the signal bandwidth

This test is aimed to find out the influence of the signal bandwidth on the convergence of the algorithm. The test is similar to that of section 5.1.1 (see Fig. 10) except the signal bandwidth which was decreased through the oversampling factor, set to r=0.4. The results in Fig. 13 show that for spectral radii less than 0.8 the number of iterations required to converge is significantly reduced as compared with higher spectral radii.

Therefore, faster convergence is achieved for lower signal bandwidth.
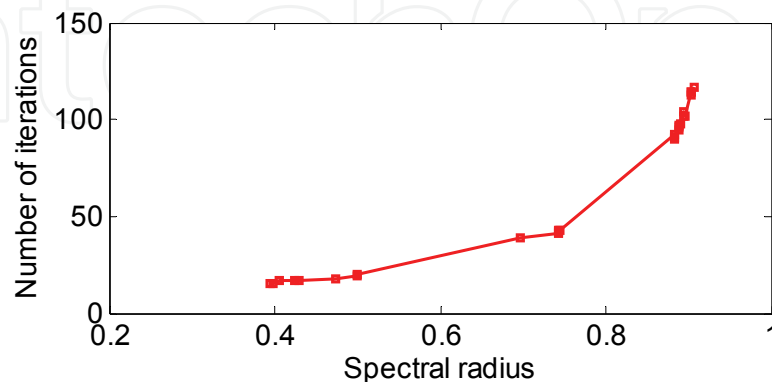
Fig. 13. Number of iterations as a function of the spectral radius (r=0.4)

Another relevant issue is to find out how the break even points are affected by decreasing the signal bandwidth. Fig. 14 shows that break even points are achieved at higher values than in the case of Fig. 12. This means that a greater percentage of missing samples is allowed in signals with lower bandwidth without reaching the non-convergence boundary.

For the interleaved geometry, the maximum spectral radius that still guarantees convergence is 0.5. Since the respective interleaving factor is m=2, then 50% of samples are allowed to be lost in this case. Comparing with results obtained in Section 5.1.1, where the signal bandwidth was greater (r=0.8), this corresponds to a significant improvement in tolerance to loss of samples. Note that in the previous case the maximum sample loss rate was just 20%. The same behaviour occurs for the random and burst error geometries. In the case of random losses, for the maximum spectral radius that still leads to a convergent situation (i.e., 0.999991), 50% of missing samples are still allowed against 13.7% in the case of r=0.8. In the case of error bursts, for the maximum allowed spectral radius of 0.999968, it is possible to have 3.9% of missing samples against 1.6% in the case of r=0.8.
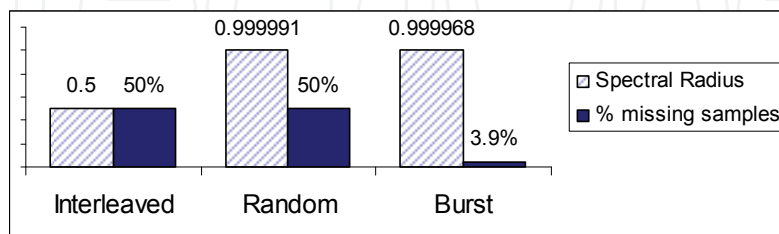

Fig. 14. Break even points for each geometry (r=0.4)

These results show that the signal bandwidth influences the convergence rate. A lower signal bandwidth leads to greater convergence rates. Also, the interleaved geometry is shown to be more tolerant to losses, which leads to the conclusion that such a mechanism is more adequate to improve error robustness and to ease signal reconstruction.

5.2 The minimum dimension interpolation algorithm

This experiments described in this subsection are intended to evaluate and compare the performance of the Papoulis-Gerchberg (PG) method with that of the minimum dimension method using both the iterative (MD Iterat) and direct computation (MD Direct) variants.

The performance metrics used in the study were the processing time obtained from Matlab[©] and the RMSE between the original and the reconstructed signals. Since the spectral radius plays an important role in the reconstruction accuracy and processing time, the dependence on the number of unknown samples was also studied.

Fig. 15 shows the dependency of the spectral radius from the percentage of missing samples for the various reconstruction methods. It is evident in the figure that the spectral radius increases with the number of missing samples, which means that in all methods more missing samples tend to result in ill-conditioned reconstruction problems. This is in line with the results of Section 5.1.2. Another important conclusion is that the spectral radius of the system matrix is independent from the reconstruction method for both oversampling factors $r=0.8$ and $r=0.6$. Moreover, it can be seen that greater bandwidth (i.e., greater r) implies greater spectral radii, which makes one to expect more processing time in the respective reconstruction. This is also in line with the conclusions of Section 5.1.3. Note that coincident lines in the figure means that for each value of r, the spectral radii are the same for all methods.
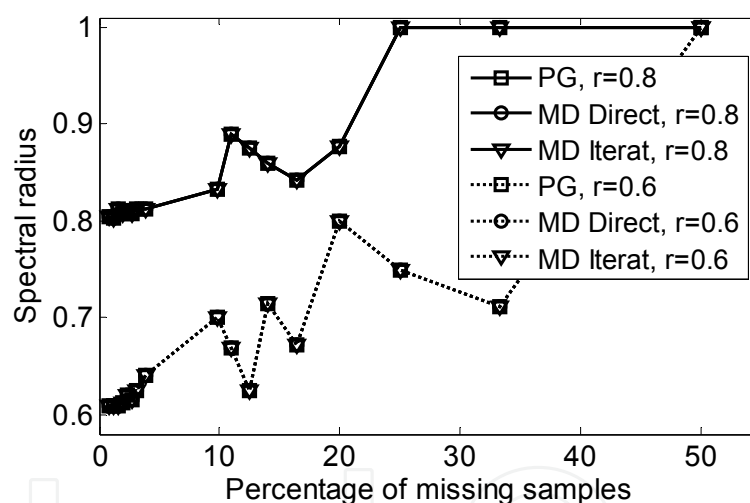

Fig. 15. Spectral radius *versus* missing samples for each method and oversampling factor

Fig. 16 shows how the RMSE between reconstructed signal and the original one depends on the number of missing samples. The break even points are also shown in the figure, separating the well-conditioning region (left side) from that of ill-conditioning (right side). In Fig. 16 one can also observe that for each oversampling factor r, both iterative methods achieve the same RMSE with the critical point occuring when the spectral radii $\rho(A)$ and $\rho(S)$ of the system matrices A and S are close to 1. $\rho(A)$ denotes the spectral radius of the maximum dimension algorithm matrix and $\rho(S)$ denotes the spectral radius of the minimum dimension algorithm matrix. For both methods, these spectral radii have the same value, $\rho(A)= \rho(S)=0.88$ corresponding to 20% of missing samples with an interleaving factor $m=5$.

Furthermore, for small percentages of missing samples, the direct computation variant (MD Direct) of the minimum dimension problem provides more accurate reconstructed signals than either maximum or minimum dimension iterative methods, i.e., the same accuracy is

obtained from both methods when the number of missing samples is low. For large number of missing samples, iterative methods exhibit slightly higher reconstruction accuracy. Therefore, when the problem is well-conditioned, direct variant computation is more suitable whereas in the case of a ill-conditioned problem, iterative methods are preferable.
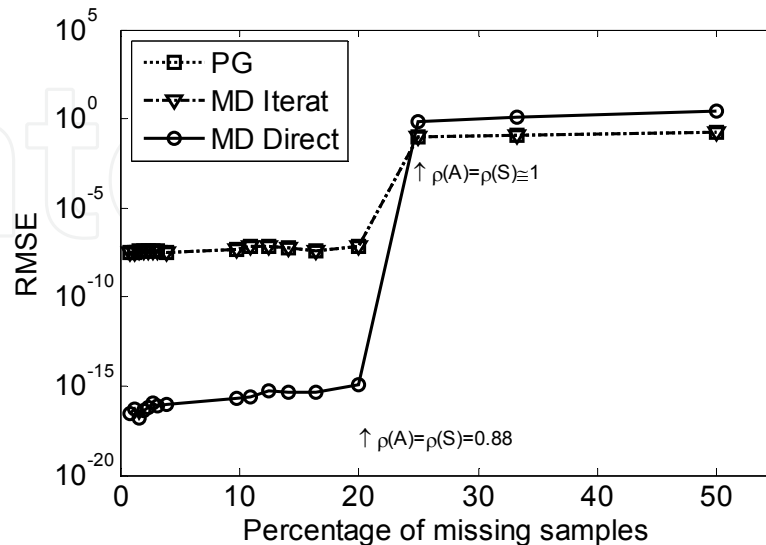

Fig. 16. RMSE *versus* number of missing samples for maximum and minimum dimension algorithms; *r*=0.8

Fig. 17 shows similar results as in Fig. 16, except that the signal bandwidth *r* is lower. The results in this figure confirm that, in the case where the number of missing samples is small, the direct variant of the minimum dimension algorithm (MD Direct) gives better reconstruction accuracy than iterative variants for both algorithms. However, for large number of missing samples, iterative variants exhibit slightly better reconstruction accuracy. The break even points are the same for both algorithms but in the figure they are shifted to the right, which means that more missing samples are allowed. In this case, it corresponds to a spectral radius of 0.71 and 33.2% of missing samples.
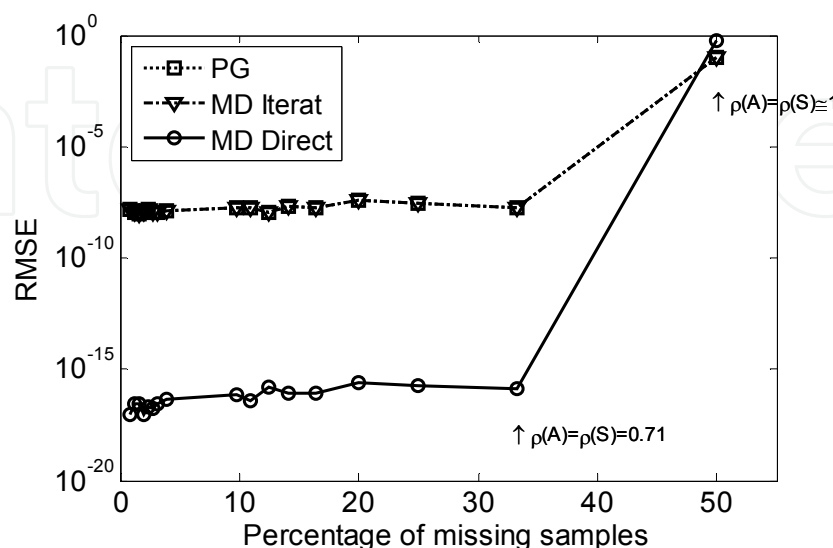
Fig. 17. RMSE *versus* missing samples; *r*=0.6

The computation time spent by the reconstruction algorithms are shown in Fig. 18 and Fig. 19, for the case of $r=0.8$ $r=0.6$, respectively. Both maximum and minimum dimension algorithms and the iterative and direct computation variants of the latter were evaluated. As it can be seen in these figures, for a small number of missing samples, direct computation of the minimum dimension problem is the fastest one and a lower bandwidth signal leads to smaller computation time, particularly when using an iterative method. However, for a large number of lost samples the direct method is more time consuming.

The processing time of the Papoulis-Gerchberg algorithm is always slower than that of the minimum dimension one, regardless of its variant, either iterative or direct computation. However the difference between them decreases when the number of missing samples increases. This is because is such case the problem dimension in the minimum dimension method approximates the maximum dimension of the Papoulis-Gerchberg.
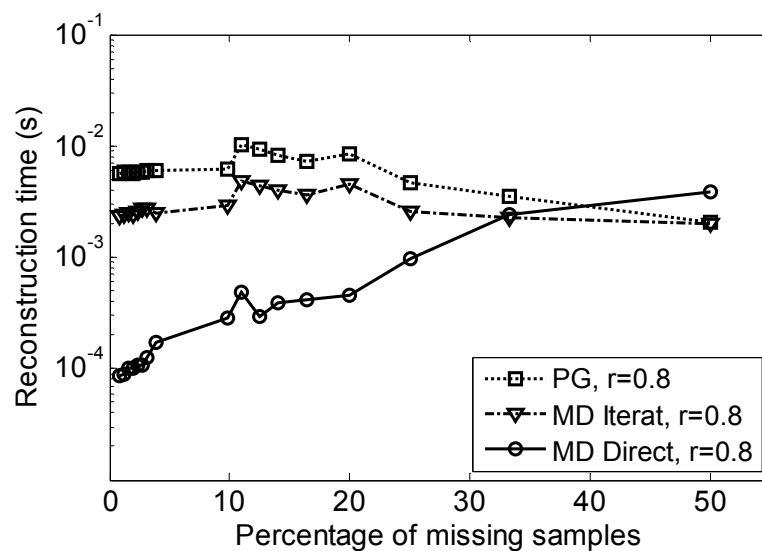
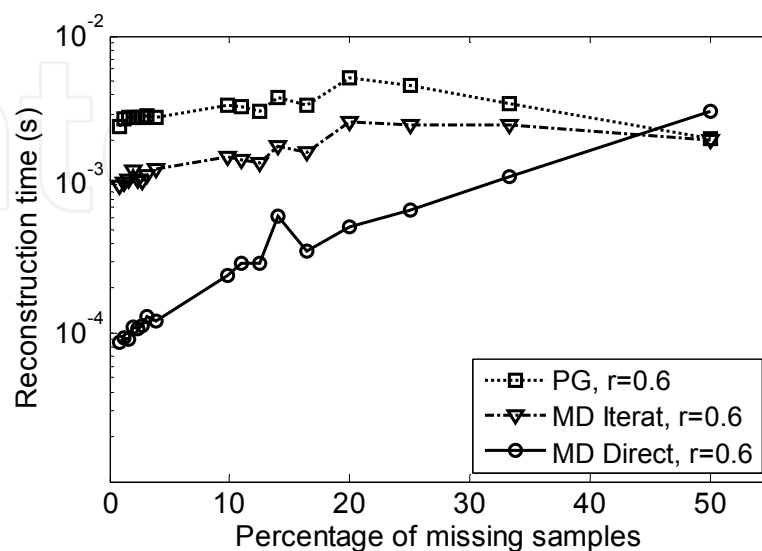Fig. 18. Computation time of reconstruction; $r=0.8$

Fig. 19. Computation time of reconstruction; $r=0.6$

6. Case study

Whilst errors and data loss increase distortion in the received voice signals, reconstruction algorithms have a significant positive impact on the voice quality. Therefore proper evaluation of the quality experienced by users is extremely important to network and service providers. The study presented in this section is part of a R&D pilot project addressing voice quality evaluation currently running at Portugal Telecom Inovação, SA (PTIn). A non-reference voice quality model was derived and validated at PTIn Labs using an IP Network and validated by using a specific probe and PESQ.

This experimental study was based on two ITU-T recommendations for voice quality evaluation: "Perceptual Evaluation of Speech Quality (PESQ)" Rec. ITU-T P.862 (ITU-T, 2001) and E-Model Rec. ITU-T G.107 (ITU-T, 2005). The E-Model was chosen as the basis for deriving the non-reference model used in the field trials, i.e., a modified E-Model.

In this trial, the impairments caused by both low bit-rate codecs and voice packet-losses of random distribution were under study. Thus, in the E-Model expression (1) ($R = R_0 - I_s - I_d - I_{e\text{-}eff} + A$), special attention has been paid to the term $I_{e\text{-}eff}$ which represents these type of impairments. The validation of the E-Model was done according to the conformance testing procedures described in the Rec. ITU-T P.564 (ITU-T, 2007a).

In the tests, the monitoring system platform ArQoS®, from PTIn, was used. This system permits to set up, maintain, monitoring and analyze telephony calls over technologies such as PSTN, GSM or IP. It provides QoS and QoE metrics such as MOS based on the PESQ algorithm. In the context of Rec. ITU-T P.564, the PESQ provides the reference for validation.

As depicted in the test scenario of Fig. 20, the main signal path includes coding and packetization, random packet-loss in an IP Network and decoding, from which the degraded signal is obtained. Thereafter, on one hand, both reference and degraded signals are given as inputs to the PESQ algorithm, whilst the output is the reference MOS used to calibrate the non-reference model. On the other hand, the degraded voice stream was collected and applied to a Gilbert modelling module whose output gives the probabilities necessary to calculate the *Ppl* and *BurstR* values for $I_{e\text{-}eff}$.
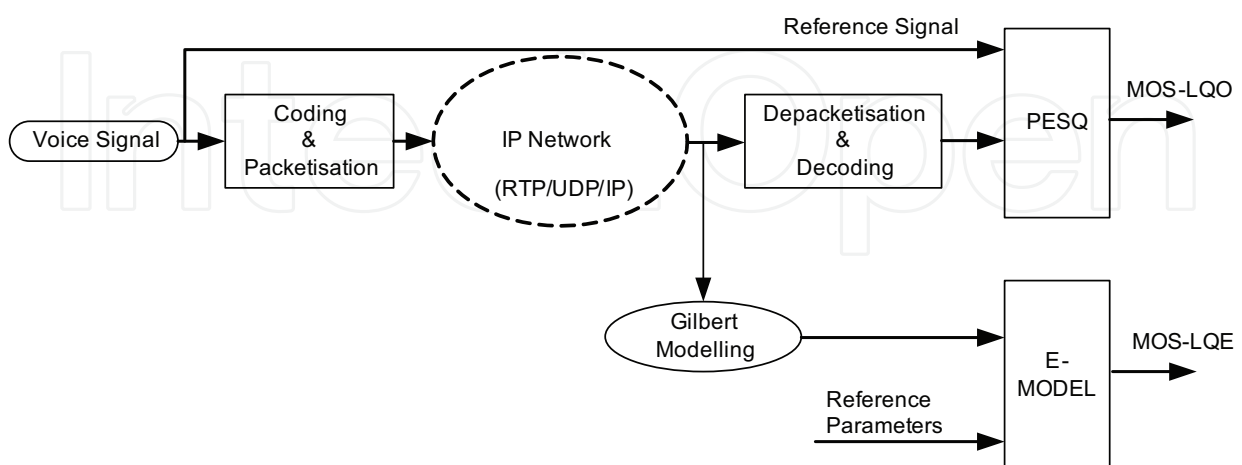

Fig. 20. Experimental setup for validation and calibration of the E-Model.

The first stage of this study aimed at achieving an accurate voice quality model based on the E-Model and using PESQ as reference for calibration. Note that both the E-Model and PESQ

are sensitive to distortions caused by codecs and packet loss. The test samples defined in Rec. ITU-T P.501 (ITU-T, 2007b) were used in the trials. Two male and two female speaker sentences were used, comprising English and Spanish languages downsampled to 8 kHz (16 bits) as required by PESQ. Table 1 shows the samples used in this calibration stage.

Test sentences	Gender	Language
These days a chicken leg is a rare dish. The hogs were fed with chopped corn and garbage.	Female 1	English
The juice of lemons makes fine punch. Four hours of steady work faced us.	Male 1	English
No arroje basura a la calle. Ellos quieren dos manzanas rojas.	Female 1	Spanish
P – siéntate en la cama. El libro trata sobre trampas.	Male 1	Spanish

Table 1. Sentences used in the first stage of the trial.

The second stage was aimed to validate the results obtained in the previous stage by using a new set of sentences and new experiments. The test scenario and the test conditions were the same as in the calibration tests described above. Table 2 shows the test sentences used in this validation stage.

Test sentences	Gender	Language
Rice is often served in round bowls. A large size in stockings is hard to sell.	Female 2	English
The birch canoe slid on smooth planks. Glue the sheet to the dark blue background.	Male 2	English
No cocinaban tan bien. Mi afeitadora afeita al ras.	Female 2	Spanish
El trapeador se puso amarillo. El fuego consumió el papel.	Male 2	Spanish

Table 2. Used sentences on the validation stage

The codecs used in the trials for evaluation and calibration were G.711, G.729 8kbps and G.723.1 6.3kbps and six average packet loss ratios were selected to take the relevant results: 0%, 2.5%, 5%, 10%, 15% and 20%. The MOS_{LQO} values obtained from PESQ, as well as those obtained from the modified E-Model were collected for each packet loss rate, codec and sentence. This results in a total of 24 tests for each codec and 24 different MOS scores for each evaluation method, i.e, the modified E-Model and PESQ. Then for each codec, regression analysis was used to calibrate the intended voice quality model. Based on these two sets of scores (PESQ and modified E-Model), the coefficients of a polynomial $p(x)$ of degree n that fits p(E-Model MOS) to MOS_{LQO} were derived.

6.1 Results and discussion

Fig. 21 shows the results obtained from regression analysis, that models the relationship between MOS_{LQO} and the modified E-Model MOS scores for G.711 codec. The horizontal axis contains the scores obtained from the modified E-Model while the vertical axis

represents the scores obtained from PESQ. For each point in the graph, the difference between the scores is the error between the modified E-Model and the reference PESQ. For instance, the second point from the left corresponds to E-Model MOS=1.5 and MOS_{LQO}=1.8, which means a MOS error of 0.3. In this case, the E-Model underestimates the MOS score in comparison with PESQ. In the graph, the points over the straight line correspond to no error cases in which both models produce the same result. In general, this figure shows that E-Model overestimates MOS relatively to PESQ. Therefore, a function to approximate the E-Model output to that of PESQ was derived. The figure shows the trend line that minimizes the RMSE between both MOS scores, which is the polynomial line that best approximates the E-Model to PESQ, for G.711 codec. Such line corresponds to the coefficients of a polynomial of degree 4 which gives the best approximation to PESQ. The resulting polynomial is given by

$$MOS_{LQO} = -0.0058 MOS^4 + 0.1252 MOS^3 - 0.6467 MOS^2 + 1.9197 MOS - 0.291 \qquad (41)$$

which is the calibrating function of the E-Model MOS in order to get the corresponding MOS_{LQO} scores.
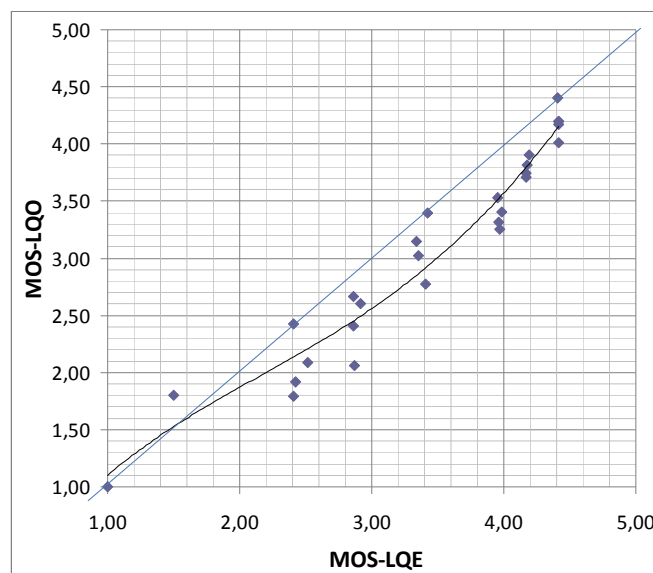

Fig. 21. Regression modelling of E-Model MOS scores as MOS_{LQO} for G.711

Fig. 22 shows the MOS scores obtained for G.729 codec under the same test conditions as in the previous case. The figure shows that in this case, the E-Model overestimates the MOS, when compared with MOS_{LQO} from PESQ. Fig. 22 also shows the trend line that best approximates the E-Model scores to MOS_{LQO} from PESQ algorithm, for G.729 codec. For this codec, the polynomial function to approximate the E-Model results to those of PESQ MOS_{LQO} is given by

$$MOS_{LQO} = 0.0554 MOS^5 - 0.7496 MOS^4 + 3.9507 MOS^3 - 9.874 MOS^2 + 11.939 MOS - 3.8293 \quad (42)$$

Finally, Fig. 23 shows the results for G.723.1 codec. In this case, the E-Model underestimates MOS, in comparison with MOS_{LQO} from PESQ. The figure also shows the polynomial trend line that best approximates the E-Model scores to MOS_{LQO} from PESQ algorithm, for G.723.1 codec.
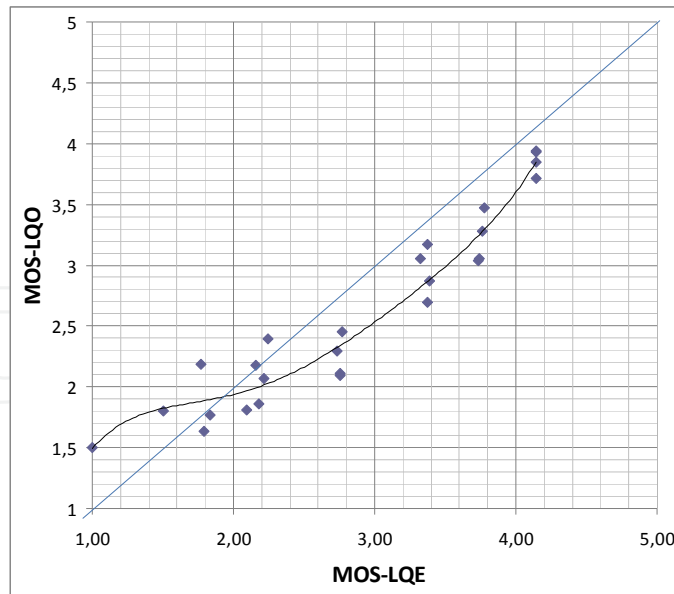
Fig. 22. Regression modelling of E-Model MOS scores as MOS_{LQO} for G.729
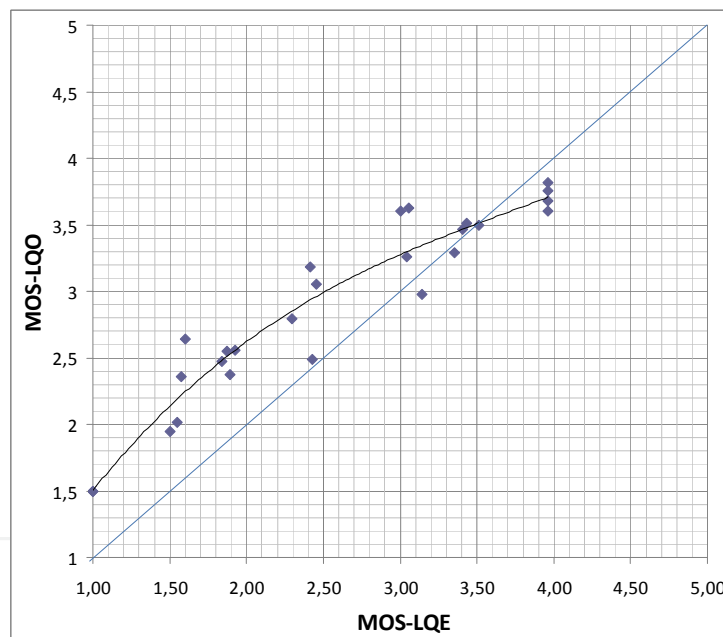

Fig. 23. Regression modelling of E-Model MOS scores as MOS_{LQO} for G.723.1

From these results, the function that best approximates MOS from E-Model to PESQ is given by:

$$MOS_{LQO} = 0.0018MOS^4 + 0.0248MOS^3 - 0.4262MOS^2 + 2.1953MOS - 0.2914 \qquad (43)$$

In the second stage, the sentences of Table 2 were used in the ArQoS® test system to obtain the respective PESQ MOS_{LQO} and E-Model MOS scores calibrated by using Equations (41), (42) and (43). Then the correlation factor, error and false positive/negative analysis between MOS_{LQO} scores and modified E-Model MOS were determined as defined in Recommendation ITU-T P.564. Table 3, Table 4 and Table 5 show the results obtained from the tests and the

conformance accuracy requirements defined in ITU-T P.564. The tables show the correlation factor, percentage of errors and false negative/false positive measures, respectively.

Measure	Results			Requirements (P.564)	
	G.711	G.729	G.723.1	Class C1	Class C2
Correlation	0.956	0.964	0.887	>0.900	>0.850

Table 3. Results for the correlation factor

Measure	Results			Requirements (P.564)	
	G.711	G.729	G.723.1	Class C1	Class C2
Quality band _B_=1 ($MOS_{LQO} \geq 2.8$)					
Errors within boundary 1 (%)	81	90	67	≥97.9	
Errors within boundary 2 (%)	100	100	100	≥97.9	
Errors within boundary 3 (%)	100	100	100		≥95.0
Errors within boundary 4 (%)	100	100	100	≥99.0	
Errors within boundary 5 (%)	100	100	100		≥97.9
Errors within boundary 6 (%)	100	100	100		≥99.0
Quality band _B_=2 ($MOS_{LQO} < 2.8$)					
Errors within boundary 7 (%)	75	86	78	≥90.0	
Errors within boundary 8 (%)	88	100	89		≥90.0
Errors within boundary 9 (%)	100	100	100	≥95.0	
Errors within boundary 10 (%)	100	100	100		≥95.0
Errors within boundary 11 (%)	100	100	100	≥99.0	
Errors within boundary 12 (%)	100	100	100		≥99.0

Table 4. Results for the percentage of errors.

Measure	Results			Requirements (P.564)	
	G.711	G.729	G.723.1	Class C1	Class C2
False negatives (%)	0	0	0	<5	<5
False positives (%)	0	0	0	<3	<3

Table 5. Results concerning false negatives/false positives

The results in Tabe 3 and Table 5, match both the correlation and false negative/false positive requirements for the Class 1. However, according to the results shown in Table 4, the percentage of errors falls within boundaries 7 and 8, which makes the modified E-Model to be included into Class 2.

Based on these results, the voice quality evaluation model based on the modified E-Model along with the respective calibration functions is currently in production at Portugal Telecom, SA.

Thus, satisfying these requirements, the voice quality evaluation model was integrated in the passive probes of ArQoS® system and is now in use at Portugal Telecom SA.

6.2 Practical application

While the ArQoS® active probes are meant to generate test calls on several type of networks, the ArQoS® passive probes are designed to analyse VoIP traffic, both signalling (SIP, Megaco, Radius, Diameter) and media stream (RTP) protocols. As passive probes, they analyse the existing traffic without any interference. They can be setup next to any element of the VoIP network, from the VoIP clients and Media Gateways to the core of the network. Collected data is gathered, analysed and processed automatically at the management system, providing many QoS statistics. The user can also use the system to trace a VoIP call in every probing point and in every protocol involved, allowing the end user to troubleshoot any possible problem.

The calibrated voice quality model of Portugal Telecom is of great use in the ArQoS® passive probes. It allows the translation of QoS metrics such as packet loss rate and jitter to a
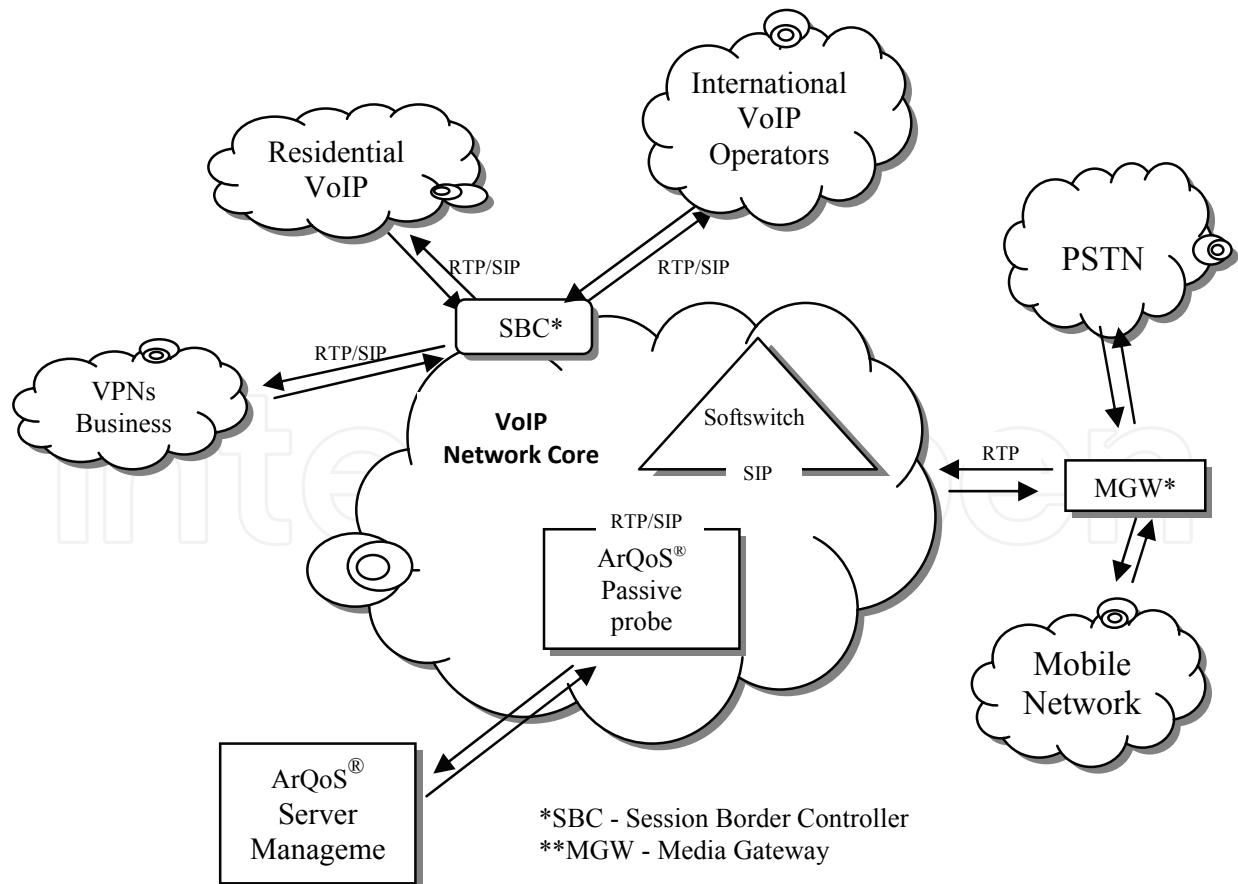

Fig. 24. Portugal Telecom VoIP network

more user friendly indicator as MOS. As depicted in Fig. 24 the ArQoS® passive probes are deployed in the Portugal Telecom VoIP Network core. All RTP streams are transmitted through the core, either in calls between VoIP and circuit-switch endpoints, or between just two VoIP clients. Our model is applied in every call then, resulting two MOS calculations, one for each way. On this application scenario, the network problems that affect the RTP stream after its passage through the core aren't really detected by the Probes. On the other hand, the reverse RTP stream that follows the same path should be affected to some extent before being analysed by the Probes. That means the user must always take into account both ways of each call. The calculated MOS values are also processed and shown in the ArQoS® statistics reporting tool, giving the users a good overview of the network voice quality.

7. Conclusion

Overall this chapter presented relevant problems of VoIP and described useful solutions, based on signal reconstruction, to overcome some of such problems. Special emphasis is given to a detailed description and comparison of two linear interpolation algorithms for voice reconstruction to cope with network errors and losses. A case study with VoIP field tests is described to evaluate the quality of VoIP services and a quality model is derived and validated.

8. Acknowledgements

9. References

Aoki, N. (2004). VoIP packet loss concealment based on two-side pitch waveform replication technique using steganography, Proceedings of *TENCON 2004 - 2004 IEEE Region 10 Conference - Analog and Digital Techniques in Electrical Engineering*, 52-55, 0-7803-8560-8 Chiang Mai, Thailand 21-24 Nov. 2004 IEEE,

Becvar, Z.; Zelenka, J.; Brada, M. & Novak, L. (2007). Comparison of Common PLC Methods Used in VoIP Networks, Proceedings of *Systems, Signals and Image Processing, 2007 and 6th EURASIP Conference focused on Speech and Image Processing, Multimedia Communications and Services. 14th International Workshop on* 389 - 392, 978-961-248-029-5, Maribor, 27-30 June 2007 IEEE,

Bhute, V. P. & Shrawankar, U. N. (2008). Error concealment schemes for speech packet transmission over IP network, Proceedings of *15th International Conference on Systems, Signals and Image Processing*, 185-188, Bratislava, Slovak Republic, 25-28 June 2008,

Cheetham, B. (2006). Error Concealment for Voice over WLAN in Converged Enterprise Networks, *IST Mobile & Wireless Communications Summit 2006*, Mykonos, Greece, 4-8 June 2006,

Erdol, N.; Castelluccia, C. & Zilouchian, A. (1993). Recovery of missing speech packets using the short-time energy and zero-crossing measurements, *IEEE Transactions on Speech and Audio Processing*, 1, 3, (July 1993), 295-303, 1063-6676

Falk, T. H. & Chan, W.-Y. (2009). Performance Study of Objective Speech Quality Measurement for Modern Wireless-VoIP Communications, *EURASIP Journal on Audio, Speech, and Music Processing*, 2009, Article ID 104382, 11 pages,

Ferreira, P. J. S. G. (1994a). Interpolation and the discrete Papoulis-Gerchberg algorithm, *IEEE Transactions on Signal Processing*, 42, 10, (Oct 1994), 2596 - 2606, 1053-587X

Ferreira, P. J. S. G. (1994b). Noniterative and fast iterative methods for interpolation and extrapolation, *IEEE Transactions on Signal Processing*, 42, 11, (Nov 1994), 3278-3282, 1053-587X

Ferreira, P. J. S. G. (1994c). The Stability of a Procedure for the Recovery of Lost Samples in Band-Limited Signals, *IEEE Transactions on Signal Processing*, 42, 11, (Nov 1994), 3278 - 3282 1053-587X

Goodman, D.; Lockhart, G.; Wasem, O. & Wong, W.-C. (1986). Waveform substitution techniques for recovering missing speech segments in packet voice communications, *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 34, 6, (December 1986), 1440-1448, 0096-3518

ITU-T (1996). Rec. P.800: *Methods for subjective determination of transmission quality* (Geneva 1996)

ITU-T (1999). Rec. G.109 *Definition of categories of speech transmission quality* (Geneva 1999)

ITU-T (2001). Rec. P.862: *Perceptual Evaluation of Speech Quality (PESQ)* (Geneva 2001)

ITU-T (2003). Rec. P.862.1: *Mapping function for transforming P.862 raw result scores to MOS-LQO* (Geneva 2003)

ITU-T (2004). Rec. P.563: *Single-ended method for objective speech quality assessment in narrow-band telephony applications* (Geneva 2004)

ITU-T (2005). Rec. G.107: *The E-Model, a computational model for use in transmission planning* (Geneva 2005)

ITU-T (2006). Rec. P.800.1: *Mean Opinion Score (MOS) terminology* (Geneva 2006)

ITU-T (2007a). Rec. P.564 *Conformance testing for voice over IP transmission quality assessment models* (Geneva 2007)

ITU-T (2007b). Rec. P.501: *Test signals for use in telephonometry* (Geneva 2007)

Jayant, N. & Christensen, S. (1981). Effects of Packet Losses in Waveform Coded Speech and Improvements Due to an Odd-Even Sample-Interpolation Procedure, *IEEE Transactions on Communications*, 29, 2, (Feb 1981), 101-109, 0096-2244

Kincaid, D. & Cheney, W. (2002). *Numerical Analysis: Mathematics of Scientific Computing*, The Brooks/Cole - Thompson Learning, 0-534-38905-08, Pacific Grove, CA, USA

Liu, F.; Kim, J. & Kuo, C.-C. J. (2001). Adaptive delay concealment for Internet voice applications with packet based time-scale modification, Proceedings of *ICASSP 2001-IEEE International Conference on the Acoustics, Speech, and Signal Processing*, 1461-1464, Washington DC, USA,

Neves, F.; Soares, S.; Reis, M. C.; Tavares, F. & Assuncao, P. (2008). VoIP reconstruction under a minimum interpolation algorithm, Proceedings of *IEEE International Symposium on Consumer Electronics, 2008. ISCE 2008*, 1-3, Vilamoura, Portugal, Apr. 2008,

Press, W. H.; Teukolsky, S. A.; Vetterling, W. T. & Flannery, B. P. (1994). *Numerical Recipes in C: The art of scientific computation*, Press Sybdicate of Cambridge University Press, 0-521-43108-5, Cambridge

Press, W. H.; Teukolsky, S. A.; Vetterling, W. T. & Flannery, B. P. (2007). *Numerical recipes: the art of scientific computing*, Cambridge University Press, 978-0-521-88068-8, Cambridge

Tang, J. (1991). Evaluation of double sided periodic substitution (dsps) method for recovering missing speech in packet voice communications, Proceedings of *IEEE Conference in Computers and Communications*, 454-458,

Tosun, L. & Kabal, P. (2005). Dynamically Adding Redundancy for Improved Error Concealment in Packet Voice Coding, Proceedings of *European Signal Processing Conf. EUSIPCO* 4, Antalya, Turkey, Sept. 2005,

Wah, B. W.; Su, X. & Lin, D. (2000). A survey of error-concealment schemes for real-time audio and video transmissions over the Internet, Proceedings of *Multimedia Software Engineering, 2000. International Symposium*, 17-24, Taipei, Taiwan,

Zourzouvillys, T. & Rescorla, E. (2010). An Introduction to Standards-Based VoIP: SIP, RTP, and Friends, *IEEE Internet Computing*, 14, 2, (March/April 2010), 69-73, 1089-7801
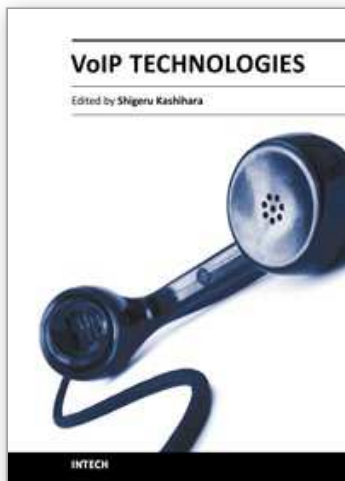
VoIP Technologies

Edited by Dr Shigeru Kashihara

ISBN 978-953-307-549-5

Hard cover, 336 pages

Publisher InTech

Published online 14, February, 2011

Published in print edition February, 2011

This book provides a collection of 15 excellent studies of Voice over IP (VoIP) technologies. While VoIP is undoubtedly a powerful and innovative communication tool for everyone, voice communication over the Internet is inherently less reliable than the public switched telephone network, because the Internet functions as a best-effort network without Quality of Service guarantee and voice data cannot be retransmitted. This book introduces research strategies that address various issues with the aim of enhancing VoIP quality. We hope that you will enjoy reading these diverse studies, and that the book will provide you with a lot of useful information about current VoIP technology research.

INTECH

open science | open minds