HMM-based channel error mitigation and its application to distributed speech recognition

Antonio M. Peinado *, Victoria Sánchez, José L. Pérez-Córdoa, Ángel de la Torre

Departamento de Electrónica y Tecnología de Computadores, Facultad de Ciencias, Universidad de Granada, 18071 Granada, Spain

Received 13 March 2002; received in revised form 6 March 2003; accepted 11 March 2003

Abstract

The emergence of distributed speech recognition has generated the need to mitigate the degradations that the transmission channel introduces in the speech features used for recognition. This work proposes a hidden Markov model (HMM) framework from which different mitigation techniques oriented to wireless channels can be derived. First, we study the performance of two techniques based on the use of a minimum mean square error (MMSE) estimation, a raw MMSE and a forward MMSE estimation, over additive white Gaussian noise (AWGN) channels. These techniques are also adapted to bursty channels. Then, we propose two new mitigation methods specially suitable for bursty channels. The first one is based on a forward–backward MMSE estimation and the second one on the well-known Viterbi algorithm. Different experiments are carried out, dealing with several issues such as the application of hard decisions on the received bits or the influence of the estimated channel SNR. The experimental results show that the HMM-based techniques can effectively mitigate channel errors, even in very poor channel conditions.

Keywords: Channel error mitigation; Distributed speech recognition; Hidden Markov models; Wireless communications; Forward–backward algorithm; Viterbi algorithm

1. Introduction

The concept of distributed speech recognition (DSR) has recently appeared as an efficient way of translating automatic speech recognition technologies to mobile and IP network applications, spurring the development of two ETSI standards for DSR (ETSI ES 201 108, 2000; ETSI ES 202 050, 2002) that have been elaborated by the Aurora working group. The underlying idea of DSR consists in using a local front-end from which the speech features are transmitted through a data channel to a remote back-end recognizer. This idea is depicted in Fig. 1. By means of this approach, only a relatively simple speech analysis is required at the local front-end, while the most important part of the computational burden is carried out at the recognition server, which can be easily upgraded with new technologies and services without any additional cost for the user. The DSR
technology also facilitates language portability, since no language information is transmitted through the data channel. In the case of mobile communications, there are additional advantages such as avoiding the effects of the speech coder and more robustness against channel errors, in comparison with the use of the speech channel.

DSR systems can be affected by several degradation sources due to the acoustic environment and the digital channel. Although the processing of all these degradations could be carried out at the receiver, the distributed nature of the recognition process in DSR makes it more convenient to treat and reduce the acoustic degradations in the local front-end, where the signal (speech plus noise) is fully available. Thus, at the remote back-end, only the errors introduced by the digital channel must be treated and mitigated.

The ETSI DSR standards include a basic mitigation algorithm that has been shown effective for medium and good quality channels on TETRA and GSM environments (Pearce, 2000). It exploits the fact that the speech parameters do not rapidly change during an error burst. In the case of lossy packet channels, several specific techniques such as interpolation (Milner, 2001) or FEC coding (Riskin et al., 2001) have been introduced for DSR. For wireless channels, exponential feature weighting has been recently proposed (Potamianos and Weerackody, 2001; Bernard and Alwan, 2001). This technique applies exponential weighting factors on the observation probabilities (those utilized by HMM models for recognition) that are computed from some reliability measure of the received bit sequences. This technique has the advantage of a direct integration of error mitigation and recognition, although this is carried out in a heuristic way.

Our work is devoted to channel error mitigation for DSR over wireless channels (although it could be extended to IP networks) based on the use of source correlations. Several concealment techniques proposed for speech coding are based on the application of a minimum mean square error (MMSE) estimation that takes into account the correlations between consecutive frames by modeling the speech source as a Markov process (Gerlach, 1993; Ligdas et al., 1997; Fingscheidt and Vary, 2001). In a similar way, the present work uses a hidden Markov model (HMM) framework to introduce several mitigation techniques for DSR. This framework contributes with two important issues. First, it provides a powerful way of introducing a priori knowledge of the speech source in the mitigation process. Also, the large theoretical background associated with HMMs can be exploited for mitigation purposes. Thus, we will show that the use of a forward–backward recursion (Rabiner and Juang, 1993) allows us to propose a new mitigation procedure based on an MMSE estimation (Peinado et al., 2001, 2002) and that the Viterbi algorithm (Rabiner and Juang, 1993) can be also applied to provide an improved decoding. The different pro-
posed techniques are developed using the Aurora front-end described in (ETSI ES 201 108, 2000), although they could be straightforwardly extended to other feature analyses and encoding schemes. It must be also noticed that the proposed mitigation algorithms affect only the decoding stage of the back-end, just before recognition, and that they do not involve any modification at the local front-end. We will test two different types of channels, additive white Gaussian noise (AWGN) and bursty channels, and also explore several aspects such as the use of hard decision on the received signal and the estimation of the channel SNR required for mitigation.

The mitigation techniques proposed in this work can be useful for other applications, different from DSR, involving speech coding. Thus, speech reconstruction from recognition features is another subject in which mitigation can be an important issue (Aurora, 2001).

The paper is organized as follows. First, we briefly summarize the DSR standard ETSI ES 201 108, its error mitigation algorithm and the Aurora framework for performance evaluation. The transmission scheme used along the paper is presented in Section 3. Section 4 is devoted to the study of MMSE-based mitigation techniques over AWGN channels, while the new techniques for bursty channels are introduced in Section 5. Finally, the conclusions of this work are summarized in Section 6.

2. DSR ETSI standard and Aurora framework

The standard ETSI ES 201 108 (v1.1.2) (ETSI ES 201 108, 2000) describes the speech processing, transmission and quality aspects of a DSR system. Although it allows three different sampling frequencies (8, 11 and 16 kHz), we will only use 8 kHz, since this is the one the Aurora-2 database (Pearce and Hirsch, 2000) uses. Frames are 25 ms long and shifted 10 ms. Each frame is represented by a 14 dimension feature vector containing 13 Mel Frequency Cepstrum Coefficients (MFCC) (including the 0th order one) plus log-energy. The first and second derivatives of the features are computed during recognition. These features are quantized using a split vector quantizer (SVQ) that groups them into pairs (MFCCs 1 and 2, MFCCs 3 and 4, ..., MFCC 0 and log-energy). Each feature pair has its own codebook that is generated using a weighted distance measure. All codebooks have a 64 centers (6 bits), except the one for MFCC-0 and log-energy, that has 256 centers (8 bits).

The bitstream is organized into a sequence of multiframes containing each a 2-octet synchronization sequence, a 4-octet header (with different informations and a multiframe counter), and a 138-octet frame packet stream which contains 24 frames grouped into pairs. Each frame pair is encoded with 88 information bits followed by a 4-bit CRC (cyclic redundancy code). The final bit rate is 4.8 kbits/s.

After decoding, an error mitigation algorithm is applied. There are two tests for error detection: a CRC checking and a data “consistency” test. This last test tries to determine whether the frames in a frame pair have a minimal continuity. When an incorrect CRC is detected, the corresponding frame pair is classified as erroneous. Besides, if the previous frame pair is “inconsistent”, it is also labelled as erroneous. From this point on, all frame pairs are classified as erroneous until one is received that passes the CRC and consistency tests. This is an effective way to detect error bursts. Once a burst, containing $2 \times B$ frames, is detected, the first $B$ frames are substituted by the last correct frame before the burst and the last $B$ ones by the first correct frame after the burst.

This standard will be compared with the different techniques introduced along the paper under different channel conditions. It must be observed that our work is exclusively concerned with error mitigation on the feature packet stream and that headers are not used for several reasons. First, the standard document does not specify how to decode them. Also, they are not necessary to carry out our experiments (assuming that the order of the received multiframes is known). Besides, it can be considered that a reliable decoding can be performed on them since the 16 header information bits are protected with other 16 parity bits.

The Aurora-2 database we have utilized in this work is based on the TI-Digits database
(connected digits) decimated to 8 kHz. We also use the speech recognizer provided by Aurora, which uses eleven 16-state continuous word HMM models (except silence and pause, that have 3 and 1 states, respectively), with 3 gaussians per state (except silence, with 6 gaussians per state, and pause, which is the central state of silence). Training is performed with 8440 clean sentences and test is carried out over the clean sentences from set A (4004 sentences distributed into four subsets). The details about the Aurora-2 experimental framework for performance evaluation can be found in (Pearce and Hirsch, 2000).

3. Transmission scheme

Fig. 2 shows a block diagram of the transmission scheme for each feature pair. As it was remarked in Section 2, the speech features are grouped into pairs that are SVQ-quantized. The resulting feature pair is represented by a vector resulting feature pair is represented by a vector 

\[ \mathbf{c} = \{ \mathbf{c}^i; i = 0, \ldots, 2^M - 1 \} \ (M = 6, 8 \text{ in this work}) \]

that, after bit mapping, is represented by a bit sequence \( \mathbf{x} = (x(0), x(1), \ldots, x(M - 1)) \ (\mathbf{x} \in \{ \mathbf{x}^i; i = 0, \ldots, 2^M - 1 \}) \), where each bit is assumed to be bipolar \( (x(k) \in \{-1, +1\}) \).

This bit sequence is transmitted, after channel encoding, through a digital channel. In the case of Aurora, a (systematic) CRC code is added to the bitstream as it is described in Section 2. However, the error correction capability of this code is very small, so it is only used for error burst detection in the Aurora standard. We will use this CRC in the same manner (only for burst detection). Therefore, the information bits are managed as if no channel encoding was applied. More details about the inclusion of channel coding are given in Section 4.

We can model the channel degradations (fading and additive noise) by means of the equation,

\[ \mathbf{y} = a\mathbf{x} + \mathbf{n} \]  

where \( \mathbf{y} \) is the received signal vector (or observation), \( a \) is a fading factor and \( \mathbf{n} \) the additive noise. The channel SNR is \( E_b/N_0 \), given that \( E_b \) is the average energy per bit (\( E_b = 1 \) in this work) and the noise variance is \( N_0/2 \).

4. Application of MMSE estimation to DSR decoding

4.1. MMSE estimation

An MMSE estimation \( \hat{\mathbf{c}} \) of a transmitted feature pair vector \( \mathbf{c} \) can be obtained as (Vaishampayan and Farvardin, 1992; Skoglund and Hedelin, 1994),

\[ \hat{\mathbf{c}} = E[\mathbf{c}|\mathbf{y}] = \sum_{i=0}^{2^M-1} \mathbf{c}^i P(\mathbf{x}^i|\mathbf{y}) \]  

By means of the Bayes rule, it can be easily derived that the \textit{a posteriori} probabilities \( P(\mathbf{x}^i|\mathbf{y}) \) can be computed as,

\[ P(\mathbf{x}^i|\mathbf{y}) = \frac{b_i(\mathbf{y})P_i}{\sum_{j=0}^{2^M-1} b_j(\mathbf{y})P_j} \]  

where \( P_i \) is the a priori probability of the codeword \( \mathbf{x}^i \) (or, equivalently, of the SVQ centroid \( \mathbf{c}^i \)), and \( b_i(\mathbf{y}) \) is the observation probability, defined as,

\[ b_i(\mathbf{y}) \equiv P(\mathbf{y}|\mathbf{x}^i) \]  

In order to perform this MMSE estimation, it is necessary to compute the observation probabilities \( b_i(\mathbf{y}) \). As suggested in Fig. 2, this computation can be carried out from the instantaneous bit error probabilities \( p_e(k) \ (k = 0, \ldots, M - 1) \) corresponding to the hard decoded bits \( \hat{x}(k) = \text{sign}[y(k)] \).

Then, the probability that bit \( x^i(k) \) was trans-
mitted given that \( y(k) \) has been received (\( \hat{x}(k) \) decoded) is,
\[
P(\hat{x}^{(i)}(k)|y(k)) = \begin{cases} 
1 - p_e(k) & \hat{x}^{(i)}(k) = \hat{x}(k) \\
p_e(k) & \hat{x}^{(i)}(k) \neq \hat{x}(k)
\end{cases}
\]
(5)

Considering bit equiprobability and a memoryless channel, it is finally obtained that,
\[
b_t(y) = C \prod_{k=0}^{M-1} P(\hat{x}^{(i)}(k)|y(k))
\]
where \( C \) is a constant independent of subindex \( i \) (it has no influence on the MMSE computation).

The bit error probability \( p_e(k) \) provides reliability information about the received data, and its computation depends on the adopted transmission scheme. Thus, in the case of a fading channel with Gaussian noise and BPSK modulation, this probability can be obtained as (Fingscheidt and Vary, 2001),
\[
p_e(k) = \frac{1}{1 + \exp(|y(k)L_c|)} \quad \text{with} \quad L_c = 4d \frac{E_b}{N_0}
\]
(7)

It must be noted that the input value \( y(k) \) is required for the computation of \( p_e(k) \), so we are implicitly assuming that soft decision is performed on the channel outputs. In the case that channel encoding is applied, the bit error probabilities must be computed in a different manner (Peinado et al., 2001; Fingscheidt and Vary, 2001). For example, the SOVA algorithm (Hagenauer and Hoehner, 1993) can estimate them in the case that the code allows a trellis decoding.

4.2. Forward MMSE estimation

The a priori knowledge about the speech source can be enhanced by modeling each feature pair generation by an ergodic continuous HMM, where each state \( s_i \) (\( i = 0, \ldots, 2^M - 1 \)) represents an SVQ centroid \( \phi^{(i)} \) (or, equivalently, codeword \( x^{(i)} \)). The HMM model is basically described by the way of computing the observation probabilities \( b_t(y) \) (developed in Section 4.1) and the transition probabilities,
\[
P(a_{ij}) = P(x_t = x^{(i)}|x_{t-1} = x^{(j)}) = P(q_t = s_j|q_{t-1} = s_i)
\]
(8)

where \( x_t \) and \( q_t \) are the transmitted bit sequence and the state at time \( t \), respectively.

We can perform now an MMSE estimation of the received parameter vector (at time \( t \)), considering the previously received signal vectors (Fingscheidt and Vary, 2001), as,
\[
\hat{e}_t = E[c[y_1, \ldots, y_i] = \sum_{i=0}^{2^M-1} c^{(i)} a_t(i)
\]
(9)
where,
\[
a_t(i) = P(x_t = x^{(i)}|y_1, \ldots, y_i)
\]
(10)
is the (a posteriori) probability that bit sequence \( x^{(i)} \) was transmitted at time \( t \) given that \( y_i \) is received at time \( t \) and that vectors \( y_1, \ldots, y_{t-1} \) have been previously received. This probability can be computed by means of the following forward recursion (Fingscheidt and Vary, 2001; Rabiner and Juang, 1993):

1. Initialization \((t = 1)\):
\[
a_t(i) = P_b(i) / K_t \quad (i = 0, 1, \ldots, 2^M - 1)
\]
(11)

2. Recursion \((t \geq 2)\):
\[
a_t(i) = \left[ \sum_{j=0}^{2^M-1} a_{t-1}(j) a_{ji} \right] b_t(y_t) / K_t
\]
\[(i = 0, 1, \ldots, 2^M - 1)
\]
(12)

where \( K_t \) is the normalization factor at time \( t \). This estimation will be referred as FMMSE (forward MMSE) in the following.

4.3. Experimental results over an AWGN channel

We test in this section the performance of the described MMSE-based concealment procedures over an AWGN channel (fading factor \( a = 1 \)). This channel provides us with information about the behavior (recognition performance) of the mitigation techniques over a uniformly degraded channel. In a later section, a more realistic bursty channel will be introduced. It is assumed that the channel SNR can be reliably estimated, so that
the exact SNR value \( E_b/N_0 \) is used in order to obtain the bit error probabilities \( p_e(k) \). The a priori and transition probabilities of the speech source have been computed from an analysis of the training data mentioned in Section 2.

The recognition performance (Word Accuracy, WAcc) results of the MMSE and FMMSE decoding procedures are depicted in Fig. 3. The performance of the Aurora decoding without mitigation (AURORA_NO_MIT) is also reported. It is not possible to establish any comparison with the Aurora mitigation procedure, since it is not designed for this type of channel, but for bursty channels, where errors are concentrated in time. Bursty channels are discussed in a later section.

As references, the word accuracies of the recognition system without channel errors are also depicted at SNR points labeled as “Base” (original features, WAcc = 99.02%) and “SVQ” (quantized features, WAcc = 99.04%). It can be observed that at a channel SNR of \(-3\) dB, the DSR system still has a reasonable behavior (more than 80% of recognition accuracy) when utilizing FMMSE, meanwhile it is severely degraded with a simple MMSE estimation. This is a proof of the importance of improving the a priori knowledge of the speech source by means of an HMM.

4.4. Use of hard decisions

As previously mentioned, we have assumed until now that soft decision can be performed on the received signal, so that it is possible to compute the instantaneous bit error probabilities \( p_e(k) \), and, therefore, the observation probabilities \( b_i(y) \). On the contrary, by means of hard decision, the received observations are directly quantized by taking the sign of the input value \((\hat{x}(k) = \text{sign}(y(k)))\), so that the computation of \( p_e(k) \) is not possible any more. When a real implementation of the decoding stage is considered, the use of hard decision implies a much simpler signal reception device. One possible solution that allows the application of the MMSE-based methods along with hard decision is to substitute the instantaneous bit error probability \( p_e(k) \) by the average bit error rate (BER) \( p_e \) in Eq. (5). Thus, Eq. (6) becomes,

\[
b_i(y) \approx C(1 - p_e)^{\frac{d(\mathbf{x}, \mathbf{x}^{(0)})}{C_p}} e^{\frac{d(\mathbf{x}, \mathbf{x}^{(0)})}{C_p}}
\]

where \( d(\mathbf{a}, \mathbf{b}) \) is the Hamming distance between codewords \( \mathbf{a} \) and \( \mathbf{b} \). In the case of an AWGN channel (BPSK modulation), the BER is computed as (Haykin, 2001),

\[
p_e = \frac{1}{2} \text{erfc} \left( \sqrt{\frac{E_b}{N_0}} \right)
\]

Thus, only an SNR estimation is required to evaluate the observation probabilities. The bit error probability of Eq. (14) can be also applied to other modulation schemes. In particular, it is a good approximation to the GMSK BER expression that applies to the GSM standard (Haykin, 2001).

The performances of MMSE and FMMSE with hard decisions (labeled as H-MMSE and H-FMMSE) are also shown in Fig. 3. An important conclusion that is extracted from those plots is that the difference between hard and soft decision is meaningfully reduced in the case of FMMSE relative to MMSE.

4.5. Influence of the SNR estimation

We have previously shown that the computation of the observation probabilities requires an
estimation of the channel SNR \( E_b/N_0 \) (see Eq. (7) or (14)). This can be a simple task in the case of channels for which the degradation is homogeneously distributed, such as AWGN or Rayleigh channels. This is not very realistic for mobile applications, where errors usually appear concentrated into bursts. Thus, in general, the SNR is variable in time and obtaining an accurate SNR estimate at each time may not be an easy task.

The following experiment has been carried out. The DSR performance is tested over an AWGN channel when an incorrectly estimated SNR is utilized for the bit error probability computation of Eq. (7). The applied mitigation is the one based on an FMMSE estimation. The results are depicted in Fig. 4. Each plot is obtained using a fixed SNR value for \( p_e(k) \) computation. Only the subset labeled as “clean1” from the set \( A \) of the Aurora-2 database is used for testing.

These plots show that a high value for the estimated SNR is more detrimental than a smaller one. In the case of the \(-4 \) dB plot, a very small degradation is obtained for high channel SNR values, while, on the contrary, an estimated SNR of \( 4 \) dB rapidly decreases the performance when the channel conditions get worse. Table 1 also shows this behavior. The best results are in the main diagonal of the table, where there is no SNR mismatch. It can be observed that the performance meaningfully decreases in each column below the main diagonal when the fixed SNR is increased. However, the degradation above the diagonal is quite small. This experiment shows us the robustness of the HMM-based mitigation when a pessimistic (small) SNR value is used. This fact will be useful later in Section 5, when we consider bursty channels.

5. Channel error mitigation over bursty channels

Very often, transmission systems must work over channels where errors are grouped into bursts. This fact must be taken into account when designing mitigation algorithms. This is clearly the case of the Aurora mitigation algorithm, which is tested in (Pearce, 2000) with three different GSM bit error patterns (EP1, EP2 and EP3) representing three different channel conditions (from acceptable to very poor quality) (ETSI TR 101 085, 2000).

The HMM framework described in Section 4 allows us to derive several mitigation techniques suitable for bursty channels. First, we will adapt the MMSE and FMMSE techniques of Section 4 to this type of channel, but also propose two new techniques based on a forward–backward recursion and the Viterbi algorithm (Rabiner and Juang, 1993), respectively (Rabiner and Juang, 1993). We will see that these new techniques have the common feature of introducing a decoding delay in order to use the correct data received before and after the burst to enhance the mitigation. This is a feature
that the Aurora mitigation procedure also has, and that we will exploit here.

5.1. A bursty channel model

We consider next a simplified bursty channel model that allows us to test the different mitigation techniques over a wide range of channel conditions. In this model, the channel noise \( n \) is obtained as the superposition of a background AWGN noise (variance \( N_g/2 \)) plus a sequence of AWGN noise bursts (variance \( N_b/2 \gg N_g/2 \)) of fixed duration \( d \) (in number of bits), with a separation given by a Poisson variable of mean \( T_b \) (Ebel and Tranter, 1995). The average variance of the channel noise is,

\[
\frac{N_0^2}{2} = \frac{N_g^2}{2} + \frac{N_b d}{2 T_b}
\]

and the average channel SNR can be computed as \( E_b/N_0 \). In our experiments, we have considered \( E_b/N_g = 6 \) dB (BER = 0.23%), \( E_b/N_b = -6 \) dB (BER = 24.59%), and \( T_b = 1500 \) bits. Thus, the different values of \( E_b/N_0 \) have the meaning of different burst durations as it is detailed in Table 2. Under these conditions, the EP3 pattern roughly corresponds to an average SNR between \( -1 \) and 0 dB (the SNR range of interest will be above these values).

5.2. Adaptation of MMSE and FMMSE estimations to bursty channels

A first step, when dealing with bursty channels, is the detection of error bursts. As we previously discussed, the Aurora mitigation algorithm performs CRC and consistency tests. We keep this detection procedure in all the experiments with bursty channels. The mitigation procedure is applied during bursts, and a standard hard-decision decoding is performed otherwise (the same as Aurora).

It must be considered that the adapted MMSE and FMMSE techniques also require an estimation of the channel SNR in order to compute bit error probabilities. This can be a difficult task for a bursty channel, for which the total SNR does not correctly describe the amount of noise at each particular burst. As we previously showed, a small value for the estimated SNR is much less damaging than a high one. Therefore, we have utilized a fixed SNR (\(-2\) dB), that is small enough for the SNR range tested in this work. This fixed SNR value will be utilized for the computation of observation probabilities during bursts in all the experiments over bursty channels. It must be pointed out that the joint use of a fixed SNR value and hard decisions involves that the observation probability computation of Eq. (13) does not depend anymore on any channel information (such as the channel SNR) but on the Hamming distance. As a consequence, our speech model becomes a discrete HMM.

We will use the notation \( Y = (y_1, \ldots, y_T) \) for the received signal vector sequence, where \( y_1 \) and \( y_T \) are the last and first correctly received vectors before and after a burst, respectively. Therefore, the sequence of erroneous vectors starts at \( t = 2 \) and finishes at \( t = T - 1 \). The forward algorithm is performed from \( t = 1 \) until \( t = T - 1 \). At the initial step (Eq. (11)), it is considered that,

\[
b_i(y_1) = \begin{cases} 
0 & \text{if } x^{(i)} \neq \hat{x}_1 \\
1 & \text{if } x^{(i)} = \hat{x}_1 
\end{cases} \quad (i = 0, \ldots, 2^M - 1) \quad (16)
\]

where \( \hat{x}_1 \) is the hard decoded codeword corresponding to the last correct received vector \( y_1 \). By means of this initialization, we are ensuring a good starting point for the forward recursion. The cases of error bursts affecting either the first or last frames of the speech utterance are considered as special cases. In case \( y_1 \) is the first vector of the speech utterance (that is, the error burst is over the beginning of the sentence), \( b_i(y_1) \) is computed by means of Eqs. (5) and (6) (or (13) in the case of hard decision). On the other hand, if the burst

<table>
<thead>
<tr>
<th>Table 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Correspondences of average SNR, burst duration (length in bits) and average bit error rate</td>
</tr>
<tr>
<td>SNR (dB)</td>
</tr>
<tr>
<td>---</td>
</tr>
<tr>
<td>-3</td>
</tr>
<tr>
<td>-1</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>5</td>
</tr>
</tbody>
</table>
affects the last frame of the utterance, the forward recursion must be continued up to \( t = T \) since this is not a correct frame.

The performance of Aurora and the modified FMMSE estimation are shown in Fig. 5. Although it is clear that Aurora reaches an acceptable performance under this type of channel, the FMMSE mitigation provides much better results. Besides, the hard version H-FMMSE obtains a performance similar to its soft decision counterpart, while the raw MMSE estimation (for both, hard or soft decision) gets quite poor results. These results provide the same main conclusion as the one obtained from the AWGN channel experiments: the use of a good speech model, such as the HMM used for FMMSE or H-FMMSE, meaningfully improves the recognition performance in comparison with the one obtained with the raw MMSE estimation. An important difference with the results obtained for the AWGN channel is that H-FMMSE reaches now results quite similar to those of the original FMMSE. This is a consequence of the concentration of errors into bursts of moderate duration that will be considered later in Section 5.5.

5.3. Forward–backward MMSE estimation

In spite of the fact that the Aurora mitigation does not reach the performance of the described FMMSE technique, it has an interesting property: it utilizes the correctly received frames that delimit a burst to rebuild the degraded frames. This idea can be translated to the MMSE estimation by considering not only the previous received vector but also the subsequent ones. Thus, we can estimate now the received feature vector at time \( t \) as

\[
\hat{c}_t = E[c_t|Y] = \sum_{i=0}^{2^M-1} c(i) \gamma_t(i)
\]

where

\[
\gamma_t(i) = P(x_t = x(i)|Y)
\]

and \( Y \) is the same vector sequence defined in Section 5.2. The conditional probabilities \( \gamma_t(i) \) can be computed as

\[
\gamma_t(i) = \frac{z_t(i)\beta_t(i)}{\sum_{j=0}^{2^M-1} z_t(j)\beta_t(j)}
\]

where,

\[
\beta_t(i) = P(y_{t+1}, \ldots, y_T|x_t = x(i))
\]

The probabilities \( z_t(i) \) can be obtained in the same way as described in Section 5.2, and the new probabilities \( \beta_t(i) \) by means of the following backward recursion:

1. Initialization (\( t = T \)):

\[
\beta_T(i) = 1 \quad (i = 0, 1, \ldots, 2^M - 1)
\]

\[
b_T(y_T) = \begin{cases} 0 & x^{(i)} \neq \hat{x}_T \\ 1 & x^{(i)} = \hat{x}_T \end{cases} \quad (i = 0, 1, \ldots, 2^M - 1)
\]

In case \( y_T \) be the last vector of the utterance (that is, the burst is over the end of the utterance), the probabilities \( b_T(y_T) \) are computed as described in Section 4.1.

2. Recursion (\( t < T \)):

\[
\beta_t(i) = \sum_{j=0}^{2^M-1} a_{ij}b_j(y_{t+1})\beta_{t+1}(j)
\]

\[
(i = 0, 1, \ldots, 2^M - 1)
\]
This recursion is performed until \( t = 2 \) except in case of burst over the first frame of the speech utterance, that it is continued up to \( t = 1 \).

It is convenient to apply at each step in the recursion a normalization factor in order to avoid underflows. We shall refer to this MMSE estimation based on the use of forward and backward recursions as forward–backward MMSE (FBMMSE) estimation.

It must be taken into account that the FBMMSE technique introduces a delay in the decoding process, since it is required to wait until the end of the burst to obtain the backward probabilities (the forward probabilities could be computed while receiving data). However, this delay is the same as the one introduced by the Aurora mitigation and is not a crucial issue in an application such as DSR.

5.4. Viterbi-based estimation

The proposed HMM framework leads us to the possibility of implementing another mitigation procedure based on the use of the Viterbi algorithm (VA) (Rabiner and Juang, 1993) that also considers the correctly received frames before and after a burst. In this case, the estimated sequence of feature pair vectors corresponds to the optimal state sequence \( \hat{Q} \) that maximizes \( P(Q, Y) \), where \( Q = (q_1, \ldots, q_T) \) represents a given state sequence.

The VA algorithm involves the recursive computation of the variable,

\[
\delta_t(i) = \max_{q_1, q_2, \ldots, q_{t-1}} P(q_1, q_2, \ldots, q_{t-1}, q_t = s_i, y_1, \ldots, y_t)
\]

Probabilities \( b_i(y_1) \) and \( b_i(y_T) \) are computed in the same way as indicated in Eqs. (16) and (22). The VA algorithm can be initiated identically to the forward procedure (Eq. (11)) and terminated by choosing the final state \( \hat{q}_T \) that corresponds to the received bit sequence \( x_T \). The whole state sequence is obtained by backtracking from \( \hat{q}_T \). Since we are obtaining a maximum a posteriori estimation of the best state sequence, we will refer to this approach as MAP estimation. This new approach presents several attractive points such as efficient implementations based on the use of trellis and log-probabilities. The delay introduced by the VA algorithm is the same as the one of FBMMSE or Aurora.

5.5. Experimental results

The effect of the Aurora, H-FMMSE, H-FBMMSE and H-MAP mitigation techniques over an error burst is illustrated in Fig. 6 for the log-energy coefficient. Meanwhile Aurora only provides a “nearest neighbor” interpolation, the HMM-based techniques can provide more accurate approximations. As it could be expected, the proposed forward–backward MMSE estimation obtains the closest plot to the original one.

The performances of the H-FMMSE, H-FBMMSE and H-MAP techniques in comparison with Aurora for bursty channels are depicted in Fig. 7. The performances of the soft-decision versions of these techniques are not shown since they obtain very similar results to their hard-decision counterparts (as shown for FMMSE in Fig. 5). H-FBMMSE provides the best results, although it involves more computational burden with respect to H-FMMSE or H-MAP. The H-MAP and H-FMMSE techniques also provide significant improvements with respect to Aurora. An additional experiment applying a linear interpolation in the cepstral domain (experiment LINEAR) (Milner, 2001) has been performed too. It is observed that

![Fig. 6. Effect of different mitigation procedures on the log-energy coefficient.](image-url)
interpolation provides even slightly worse results than the frame repetition method implemented in the Aurora standard.

We have also tested the H-FBMMSE, H-FMMSE and H-MAP techniques with the EP GSM error patterns (ETSI TR 101 085, 2000). In this case, the degradation is introduced in the signal vectors by means of an error pattern $e$ as,

$$y = x \oplus e$$  \hspace{1cm} (25)$$

where $e$ is chosen among EP1, EP2 and EP3 (previously decimated for their use on a 4.8 kbits/s data channel). The results are shown in Table 3. For patterns EP1 or EP2, the degradation is negligible for all the tested techniques. However, when the EP3 pattern is applied, the Aurora mitigation introduces more than 5% of word accuracy reduction, while the H-FBMMSE technique yields less than half a point of degradation. Also, the H-FMMSE and H-MAP techniques show quite a good behavior, since both introduce a degradation of less than 1% of WAcc with respect to H-FBMMSE, with the H-MAP procedure performing slightly better than H-FMMSE. Again, a linear interpolation mitigation provides worse results than Aurora.

<table>
<thead>
<tr>
<th>$\text{WAcc (%)}$</th>
<th>EP1 ($\text{BER = 0%}$)</th>
<th>EP2 ($\text{BER = 1.76%}$)</th>
<th>EP3 ($\text{BER = 3.48%}$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{Aurora}$</td>
<td>99.04</td>
<td>98.94</td>
<td>93.40</td>
</tr>
<tr>
<td>$\text{H-FBMMSE}$</td>
<td>99.04</td>
<td>99.01</td>
<td>98.66</td>
</tr>
<tr>
<td>$\text{H-FMMSE}$</td>
<td>99.04</td>
<td>99.01</td>
<td>97.96</td>
</tr>
<tr>
<td>$\text{H-MAP}$</td>
<td>99.04</td>
<td>99.00</td>
<td>98.09</td>
</tr>
<tr>
<td>$\text{Linear}$</td>
<td>99.04</td>
<td>98.96</td>
<td>92.87</td>
</tr>
</tbody>
</table>

### 6. Summary

We have proposed several methods for channel error mitigation in a DSR application that are derived from an HMM framework. First, we have studied the behavior of the raw MMSE and FMMSE estimations, based on soft decisions on the channel outputs, over an AWGN channel. The second estimation method can provide a much better performance than the first one by considering previously received vectors for the estimation of the current feature vector. In order to do this, the a posteriori probabilities required for the MMSE computation are obtained from a forward recursion. Then, we have introduced a hard decision approach that avoids the soft decision requirement, with a small loss of performance in the case of FMMSE, and have shown the robustness of FMMSE when a small SNR value is used for the computation of the observation probabilities.

In order to test the behavior of the different MMSE estimation techniques under more realistic conditions, we have also checked the use of a bursty channel model. In this case, the FMMSE technique requires some adaptation to this type of channel. This adapted FMMSE meaningfully improves the performance given by the Aurora mitigation algorithm. Also, the use of hard-decision on FMMSE does not involve any loss of performance in this case. The idea of Aurora of utilizing the correctly received frames before and after the error burst to carry out the mitigation is also incorporated to the MMSE estimation, which is performed by means of a forward–backward procedure (FBMMSE technique). This idea is also
exploited by utilizing the MAP estimation of the received sequence of feature vectors that it is obtained when a Viterbi decoding is applied. All the considered techniques (Aurora, FMMSE, FBMMSE and MAP) were compared over a bursty channel, applying hard decisions. As expected, FBMMSE obtained the best results. Finally, the FMMSE, FBMMSE and MAP estimations (also with hard decisions) were compared with Aurora and linear interpolation applying the GSM error patterns EP1, EP2 and EP3 to the coder output. The most noticeable difference among them is observed with the EP3 pattern, for which our techniques obtain an absolute word accuracy improvement between 4% and 5% with respect to Aurora. We must point out that for the application of these hard decision versions only the decoded bit vectors and the boundaries of the error bursts (that is, the information that the Aurora ETSI standard provides) are required, since no SNR estimation is needed (a small and fixed value is enough to obtain quite acceptable results).

All the obtained results have confirmed the power of the proposed HMM framework in an application such as DSR. The selection of a specific mitigation technique among the proposed ones depends on several factors to be considered such as computational burden or the maximum allowed delay.

Acknowledgements

We would like to thank David Pearce (from Motorola Labs) for providing us with the GSM EP patterns.

References


ETSI TR 101 085 v8.0.0, 2000. Digital cellular telecommunications system (Phase 2+) (GSM); performance characterization of the GSM enhanced full rate (EFR) speech code. June 2000.


