Speech Recognition in Mobile Environments

Juan M. Huerta

Department of Electrical and Computer Engineering
Carnegie Mellon University
Pittsburgh, PA 15213

Submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

April, 2000

Abstract

The growth of cellular telephony combined with recent advances in speech recognition technology results in sizeable potential opportunities for mobile speech recognition applications. Classic robustness techniques that have been previously proposed for speech recognition yield limited improvements of the degradation introduced by idiosyncrasies of the mobile networks. These sources of degradation include distortion introduced by the speech codec as well as artifacts arising from channel errors and discontinuous transmission.

In this thesis we focus on characterizing the distortion introduced to the speech signal by the speech codec and we propose methods for reducing the detrimental effect of coding on recognition accuracy. The initial focus of this thesis is on the full rate GSM codec (FR-GSM). We propose a method to generate recognition features directly from codec parameters. It is shown in this work that by selectively constructing a cepstral feature vector from the GSM codec parameters it is possible to reduce the effect of coding on recognition.

The later parts of this work are related to weighted acoustic modeling for robust speech recognition. The motivation for this approach is based on the observation that not all phones in a GSM-coded corpus are distorted to the same extent due to coding. We first establish a set of phonetic distortion classes through an analysis of the distribution of the log spectral distortion introduced to each phone by the codec. These classes are then employed to estimate an optimal weighted combination of acoustic models according to the average distortion encountered by the class. A relative reduction of almost 70% of the degradation introduced by the GSM codec was achieved using this method.

The technique of weighted acoustic modeling based on instantaneous distortion is introduced as an alternative to the method based on average distortion information. When the extent of cepstral distortion introduced by coding is known, weighted acoustic modeling provides a reduction of about 50% in the word error rate introduced by concurrent GSM and CELP. We propose two methods to estimate the instantaneous distortion information: one based on recoding sensitivity and another based on long-term predictability. Due to the non linear relation between the time and the log-spectral domain, the proposed estimates of the instantaneous distortion do not perform as well as algorithms based on knowledge of cepstral distortion. However, we show that employing the proposed instantaneous distortion information estimates can help obtain the best recognition results established in the baseline conditions employing only 50% of the baseline Gaussian density computations.

Acknowledgements

First I would like to acknowledge the help, support, direction and advice of Richard Stern. This work has substantially benefited from his guidance and input. I would also like to thank the other members of the thesis committee, Prof. Vijayakumar Bhagavatula, Prof. Tsuhan Chen and Prof. Hynek Hermansky for their comments and feedback.

I am indebted to the whole current and past members of the SPHINX group, and the Robust Speech Recognition group for the great intellectual infrastructure they have accumulated at CMU through the years. At CMU, I have learned and been inspired from its thriving speech community; in particular I would like to thank Pedro Moreno, Eric Thayer, Ravi Mosur, and Daniel Tapias (from Telefónica TI+D). Particular mention should also go to Rita Singh, and Bhiksha Raj for their feedback and proofreading contributions. In addition, Pedro, Evandro, Matt, Sam-Joo, Uday, Jon, and Mike have constituted a great team all these years and have demonstrated that good work and great camaraderie are not mutually exclusive. Thanks also to all the administrative, faculty and student community of the ECE Department; they really constitute a world class environment for research, learning and fun. Thanks also to the great people at Dragon Systems, I am proud of having been a Dragon myself once

I would also like to acknowledge and thank the support and nurturing that has come all these years from my friends and family in México, and friends in the United States and Pittsburgh. Special thanks to my mother, siblings and cousins, for providing me with the spiritual and emotional resources and encouragement to go on. Thanks to all my friends in México that never hesitate to bring me back into the rhythm of things. Thanks to all the friends in Pittsburgh: both at CMU, at Pitt's LRDC. Particular thanks to Despina, my endless source of inspiration, tireless confident and reliable oasis in times of tribulation.

in memoriam
R.P. Rogerio Carranco Guzmán

Table of Contents

List of Figures

List of Tables

Chapter 1
Introduction

Recent progress in Automatic Speech Recognition (ASR) technology has enabled the development and deployment of more sophisticated and more accurate speech recognition applications. This progress, combined with an explosion in the capabilities and use of wireless and mobile communication and computing terminal devices, makes it feasible for ASR to become a common feature and service for current and future portable terminal devices and in mobile or wireless networks. This mobile ASR capability can be applied both as a user interface to the terminal device as well as a data Input/ Output modality between the user and the remote application.

Due to the versatility and diversity of the capabilities and characteristics of these devices and networks, it is expected that various modalities (or modes of operation) of speech recognition will exist in third-generation mobile environments. The modalities of ASR that we refer to, imply that mobile speech recognition can be characterized by the location where the recognition takes place; for example, recognition can take place in the terminal device, in a central server, or in a mixed or distributed scenario. The constraints that the network imposes on the bit rate of the transmitted signal, the limitations imposed by the computing capabilities of the device on the complexity of the signal processing front-end and the decoder, compounded with the potential exposure of the user to more intense and challenging acoustic environments, make the problem of ASR in mobile environments more susceptible to performance degradation than fixed network speech recognition applications.

In previous efforts by other researchers, significant work has been devoted to the problems of acoustic robustness in the fixed telephone network (*e.g.*, [17, 29, 58]) and robustness under linear channel distortion and additive noise (*e.g.*, [1, 60]). In recent years, a significant effort has been made to focus more closely on the problems that are idiosyncratic to recognition in mobile and cellular environments (*e.g.*, [12, 59, 73]).

Due to the expected explosion in the use of wireless devices to access the internet [63, 64], it is expected that the interest in deployment of mobile speech and voice applications will continue, and that this interest will fuel further research in Robust mobile ASR considerably. This dissertation aims to contribute in the continuous efforts to improve recognition in mobile ASR applications.

In this dissertation we explicitly focus in the problem of recognition through a wireless digital communication network; such a network requires considerable reduction of the bit rate of the speech signal in order to economize bandwidth. By representing the speech signal employing a significantly smaller number of bits, this coding process introduces distortion to the reconstructed speech signal that deteriorates the performance of the speech recognizer.

In this work, we will focus particularly on the operation and the effect of the Full Rate GSM codec on the recognizer. Later in this thesis, we generalize the techniques we develop for GSM coding to other type of codecs using Short-term/Long-term analysis, which includes Linear Predictive Analysis-by-Synthesis type of codecs (LPAS). To verify the extensibility of some of our ideas, in the final parts of this thesis we will evaluate our techniques with a different codec standard (the FS-1016 codec) that operates at a different bit rate.

In the remainder of this chapter we present an introduction to the problem of speech recognition in mobile environments with emphasis on the communication paradigm we will be working on and an outline of the rest of this thesis.

1.1 Modalities of mobile speech recognition

In this section we enumerate the characteristics and limitations that ASR applications encounter under the different modalities in which they can be deployed employing wireless digital communication links. As we have mentioned in the beginning of

this chapter, mobile speech recognition can be characterized according to the location where recognition takes place. This defines three principal modalities: (1) network speech recognition, (2) terminal speech recognition, and (3) distributed speech recognition. Each of these modalities can have very different and characteristic effects on the performance of ASR systems. We describe these modalities in more detail:

1.1.1 Mobile network speech recognition

In this modality, the recognizer is implemented in a location remote to the user so the speech signal has to be transmitted from the user's terminal to the recognition server through a wireless link. The most evident contrast of this modality with respect to fixed network recognition (*e.g.*, telephone-based recognition) is the wireless channel transmission medium. The implications of this basic difference are twofold: the need for reduction of the signal's bit rate through source coding techniques and the effects of the wireless transmission channel on the reconstructed signal (*i.e.*, transmission errors, data dropouts, interference noise etc.). However, having a recognizer residing in a central server enables larger and more powerful computers to perform recognition, permitting more sophisticated and elaborate ASR applications (*e.g.*, dialog-based systems which currently incorporate parsers, natural language modules, speech synthesis and database queries) than those possible on terminal devices.

1.1.2 Mobile terminal speech recognition

In this modality, the recognition is performed in the user's terminal device. In this case, the speech signal does not travel through a wireless communication network, so it is unaffected by the transmission channel or source and compression algorithms. However, computational and memory resources often have to be constrained due to the cost-sensitive nature of the terminal devices, making only relatively simple recognition systems currently possible (*e.g.*, hands free voice dialing, basic command and control applications etc.).

1.1.3 Distributed speech recognition

In this modality the ASR application processing and computation routines are distributed between the terminal and the central ASR server. This allows for recognition not to depend on a speech signal that has been affected by the wireless network channel and compression and coding. Instead, a typical scenario of this modality involves the following configuration: the feature parameters are extracted at the terminal device and transmitted as data, possibly through an error-protected channel, to a network-based recognizer. As Haavisto points out [33], the main disadvantage of this approach is the dependence on a standardized front-end. Establishing and standardizing such a front-end involves hard problems to be solved as the recognition features should allow high accuracy recognition for clean environments, yet be robust to noise. A standardized front-end will also need to consider multi-linguality, the Lombard effect, gender robustness, etc. It will also have to be device and microphone-independent in order to minimize the impact of terminal equipment variability on the recognizer's performance. There exists an ongoing standardization effort at the European Telecommunications Standards Institute (ETSI) that seeks to establish such standards. A system of this type can benefit from the advantages of the two modalities we have previously described: sophisticated systems can be implemented (as in mobile network ASR), while the features are computed and possibly normalized and compensated at the terminal level (as in mobile terminal ASR).

1.2 Issues common to the mobile speech recognition modalities

The three mobile speech recognition modalities differ in broad terms in terms of whether or not the transmitted coded speech is used for recognition, and whether the recognizer resides on the terminal device. In spite of this dissimilarities, these modalities share the following issues:

- **Potential exposure to intense environmental noise**: This problem is compounded by the fact that the additive noise might be highly non-stationary. Speakers may

also modify, albeit unintentionally, their speech characteristics when speaking under intense noise conditions (the Lombard effect). Another problem common to hand-held devices that affects the quality of the speech signal is the physical placement of the terminal device. These type of distortions in the signal typically affects recognition substantially.

- **Terminal equipment devices are cost sensitive**: This implies that terminal devices will allow only limited computational capabilities and thus allow relatively limited front-end signal processing, feature extraction or recognition algorithms. Thus, the mobile network, terminal, and distributed speech recognition modalities will have to rely, at least in the immediate future on relatively simple signal processing, front-ends, and terminal recognizers, as well as inexpensive microphones. This also implies that in the immediate future, given current ASR technology and computational resources, complex ASR applications such as dialog systems will only be possible to implement in the network recognition modality, as they rely on sophisticated recognizers and compensation routines, which are currently feasible only with relatively large central speech servers.

1.3 Topology of a mobile network speech recognition application

Figure 1.1 shows a diagram of an ASR application linked to a user through a mobile network linked to a land-line or Public Switched Telephone Network (PSTN) network, based on the mobile topology presented by Calhoun [9]. The mobile terminal communicates to a radio base station through a wireless channel. In order to be transmitted through the wireless link the speech signal is coded, modulated, and transmitted. In order to achieve this signal source coding, digital cellular networks digitize the reduced bandwidth speech (300 to 3.2 khz) and represent it at a considerably lower bit rate (currently between 3 and 13 kbps) by performing a LPAS type of coding [48, 49]. The coded speech gets organized into packets of bits (called frames), and modulated

and transmitted to the cell site. In the cell site the packages are demodulated and the speech is decoded. At this point it is possible that errors have been introduced in the bit-stream representing the signal due to interference noise in the transmission. Current channel coding standards have provisions for the detection of frames containing these type of errors. For a speech signal, it is possible to establish the magnitude of the perceptual impact that errors on the different individual codec parameter bits will produce. Based on this type of analysis, the most sensitive bits are protected or coded more robustly (*e.g.*, using more bits) than bits carrying less "perceptually important" information. Also, wireless standards permit the regeneration of distorted frames by the extrapolation of speech codec parameters from adjacent undistorted frames exploiting the correlation or continuity that exists in the speech signal between adjacent frames. If the data loss or damage is determined to be large enough such that data extrapolation would not produce a reasonable restoration of the signal, some standards have provisions for fading out the signal and inserting artificial white noise.

The radio base station is connected to the radio base station controller through dedicated voice grade circuits. Between the radio base station controller and the ASR application, the speech signal needs to get transferred to the PSTN network. Even though a second source coding process exists typically in this link, this processing is not considered to affect the ASR performance significatively because of the higher bit rates of these codecs. After this process is performed and the speech signal is transferred to the PSTN network, the speech signal arrives in analog mode to the speech recognition application.

One of the most significant sources of compression in the whole process might be the processing and compression that the speech signal needs to undergo in order to be transmitted through the wireless link; we will refer to this speech coding simply as "coding". As we described above, there may be other codec standards implemented in the PSTN or in the links between the different offices, controllers and switches. However, these coding stages normally have considerably higher bit rates and will not represent a distortion or degradation of importance in the link. The effect of having

several speech codecs acting on the speech signal in series is called tandeming. Some codecs present the property of synchronous tandeming [11], meaning that no further distortion is introduced to the signal if two or more of these codecs act on a signal. Digalakis [18], has studied the effect of codec tandeming in ASR for various coding topologies that might exist between the terminal and the ASR application.

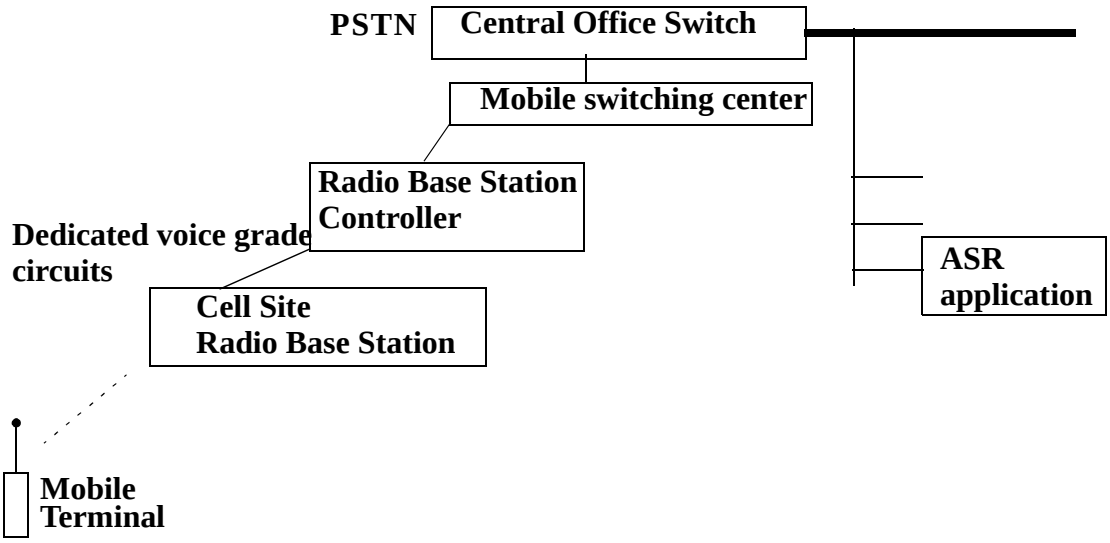

Figure 1.1 Diagram of an ASR application connected to a mobile terminal through a mobile and PSTN connections.

1.4 Scope and organization of this thesis

The use of different optimization criteria in the design of speech coding algorithms and in the development speech recognition systems and selection of features for ASR results in the suboptimal performance of recognizers when recognizing coded speech. In this dissertation we focus on the problem of transmission-error-free *mobile network speech recognition.* Specifically, in this work, our aim is to focus on the operation principles of the GSM codec; based on an analysis of the effects of quantization and coding on the components speech signal we will improve ASR robustness. We will

also extend our analysis and findings to the FS-1016 codec, which is a CELP-based type of codec.

More specifically, we will see that most modern commercially-implemented speech codecs are designed around the short-term/long-term predictability analysis of the speech signal. The goal of a speech codec is to remove the redundancies of the speech signal; short-term analysis removes the short-term redundancies of the signal while long-term analysis removes the long-term ones. Speech recognition normally does not benefit from or gets affected by the information carried in the long-term component of the speech signal (the pitch related parameters), and traditionally such information is discarded prior to recognition. In other words, current ASR front-ends have been designed to retain short-term information only. However, we will show that ASR does indeed get affected by the distortion introduced by the codec when attempting to represent this long-term information. Because of the importance of the pitch information and in general of the short-term residual signal to be able to faithfully represent the original speech signal, speech codecs reserve a large percentage of the bits available to represent the residual signal. Nevertheless, the reconstructed residual signal contains a large amount of quantization distortion in spite of this allocation of a large percentage of the available bits. At the front-end level, we will study the impact of coding on the ASR feature and we will see if it is advantageous to circumvent the quantization introduced to the short-term residual. When recognizing from the reconstructed speech signal, we propose to exploit the time-varying distortive behavior of the codec under changing long-term predictability conditions of the signal in order to achieve robust ASR performance under coding. Our approach to ASR robustness will exploit the above-mentioned properties of the coding process in the context of HMM weighted acoustic modeling.

The organization of the thesis is as follows. In Chapter 2 we present a brief overview of the problem and present some ASR background. We also describe our research environment and present some baseline results using off-the-shelf robustness techniques. In Chapter 3, we contrast the analysis performed for feature computation for

speech recognition with the analysis performed in speech coding. We also present brief descriptions of the operation of the two codecs that we will be using later on, the FR-GSM codec and the FS-1016 codec. Chapter 4 presents a more detailed description of the effects of GSM coding and quantization of the speech codec parameters on the signal, and provides an analysis of the origin of such distortion. We also introduce here the technique related to deriving cepstral features for recognition directly from GSM codec parameters, thereby merging the ASR front-end with the GSM codec. In Chapter 5 we associate the concepts of long-term predictability and phonetic properties with the effect of the speech codec on the signal, we also relate these effects to their impact on recognition accuracy. Chapter 6 introduces the concept of multiple weighted acoustic models for robust ASR under speech coding, based on the observations made in Chapter 5. Chapters 7 and 8 apply the idea of weighted acoustic modeling based on average phonetic distortion (Chapter 7) and based on the instantaneous estimation of the Relative Log Spectral Distortion (RLSD) introduced by the codec (Chapter 8). In Chapter 8, we also introduce the problem of ASR in the context of multiple concurrent codecs, for which we propose applying long-term predictability-based information. Chapter 9 presents the conclusions of our findings as well as some ideas and directions for future work in this area.

Chapter 2
Automatic speech recognition in a mobile network environment

In this chapter we present an overview of the HMM based approach to speech recognition; we also enumerate the sources of degradation that a speech recognizer will encounter when in the context of a mobile network application. This chapter also details the system configuration and databases with which we will be working, and establishes a series of recognition baselines that we will later on use as reference. We finally present a brief overview of work done by others in the area that can be considered related or relevant to this work.

2.1 A mathematical formulation of the speech recognition problem

The problem of automatic speech recognition consists of the task of transcribing the associated text of an utterance. This task is performed given an observation of the features of a realization of the utterance. The most successful and widely used approach to speech recognition is of a statistical nature and is based on Hidden Markov Models (HMMs) [5, 44, 45] which we briefly outline here.

Let A be the observed set of feature vectors computed by the front-end processor of the recognizer from the speech signal. Let this sequence of feature vectors be denoted by $a_1, a_2, a_3, \ldots, a_m$, which, without loss of generality can be thought of as symbols taken from a possibly very large alphabet Ψ:

$$A = a_1, a_2, a_3, \ldots, a_m \qquad (a_i \in \Psi) \tag{2.1}$$

Let the utterance W produced by a speaker be a sequence of n words belonging to a fixed given vocabulary ϑ:

$$W = w_1, w_2, w_3, \ldots, w_n \qquad (w_i \in \vartheta) \tag{2.2}$$

If $P(W|A)$ denotes the probability that the string of words W was produced given the acoustic observation A, then a speech recognition system will have the objective of finding the most likely word string $\hat{W}$ given the observed acoustic evidence A:

$$\hat{W} = \underset{W}{\arg max} \; P(W|A) \tag{2.3}$$

Strictly speaking, we can implement a recognizer based on the above idea. But this sort of approach (called direct modelling), implies the need to have an inventory of all the possible acoustic realizations A of every single utterance, which is infeasible. To make the problem tractable we base our recognition in terms of $P(A|W)$, which can be modeled more easily than the case above following a source-channel model [45] and phonetic modeling of the string sequences W.

Using the Bayes' formula, the expression above can be rewritten in terms of $P(A|W)$, that is, in terms of the probability of obtaining an acoustic observation A given that the string W was uttered,

$$\hat{W} = \underset{W}{\arg max} \; \frac{P(W)P(A|W)}{P(A)} \tag{2.4}$$

The term $P(A)$ is the average probability of observing A and is not a function of the uttered string W, so it can be eliminated from Equation 2.4. The problem now

reduces to the maximization of the term $P(W)P(A|W)$. The term $P(W)$ represents the probability of the string W, regardless of acoustic observations. This term can possibly be computed considering the semantic, syntactic, grammatic and pragmatic likelihood of the candidate string W. The term $P(A|W)$ is evaluated using a set of acoustic models that describes the likelihood of the observed feature sequence for every candidate hypothesis. Modern speech recognizers perform this acoustic modeling making use of a phonetic representation of the words in the system's dictionary, and a collection of Hidden Markov Models associated to each phonetic unit of the task's language. We describe the HMM approach to acoustic modeling in the next subsection.

2.1.1 Components of a frame/HMM based speech recognition system

In order to find the most likely word string $\hat{W}$ given the acoustic observation A in Equation 2.4 above, modern automatic speech recognizers evaluate the term $P(W)$ through the means of a statistical language model component and the term $P(A|W)$ through an HMM based acoustic model component. Even though the vector sequence A, can be a represented in many different feature domains (*e.g.*, cepstral vectors, LPC vectors, PLP vectors, segment modeling, formant trajectories), and the features chosen for recognition is tightly linked to the assumptions made for acoustic modeling, the HMM approach to speech recognition is largely independent of the feature chosen to represent the speech signal. We briefly describe the three components of a frame-based HMM speech recognition system:

- **Acoustic features:** The typical set of features derived from the speech signal is the cepstral vector. Cepstral frame-based systems typically generate features at a rate of 100 frames per second, spanning a segment of speech with duration of approximately 25 msecs. This means that the features will have a certain overlap and thus cannot be completely conditionally independent. In addition to the cepstral features, typical feature vectors include first and second order time derivatives of the cepstral information. These cepstral time derivatives carry information related to

the time varying properties of the signal, and have been found to enhance recognition considerably [35]. In addition, they help to make the observations more conditionally independent. A typical cepstral front-end, also performs a human-perception based scaling of the frequency axis based on perceptually derived warping functions (*e.g.*, mel scale, Bark scale). When cepstral features are derived from spectra warped according to a mel scale, we refer to them as mel scale cepstral coefficients, or MFCCs. Cepstral features based on perceptual linear prediction [36] are also commonly used; this type of features also attempt to incorporate perceptual properties of the human hearing apparatus to the feature representing the speech signal. Chapter 3 further elaborates this topic.

- **Language Modeling:** In order to obtain the prior probability $P(W)$ of every candidate string W in Equation 2.4 above, we can apply the definition of conditional probability to restate the language model expression as:

$$P(W) = P(w_1, w_2, \ldots, w_n) = \prod_{i=1}^{n} P(w_i | w_1, \ldots, w_{i-1}) \tag{2.5}$$

In order to obtain estimates of the word history terms $P(w_i | w_1, \ldots, w_{i-1})$ keeping the problem tractable, we employ equivalence classes $\Phi(w_1, \ldots, w_{i-1})$; then the problem of language model consists in the determination, and evaluation of appropriate equivalence classifications and word history probabilities. Typically, recognizers are based on trigram and bigram language models: histories are the same if they end in the same two words or one word, respectively. Then, a trigram language model becomes,

$$P(W) = P(w_1, w_2, \ldots, w_n) = \prod_{i=1}^{n} P(w_i | w_{i-2}, w_{i-1}) \tag{2.6}$$

- **HMM based Acoustic Modeling:** To compute the probability of the observation sequence A given the word sequence W, we construct sequences of states (or chains of states) in which we concatenate subchains representing the phones that compose the pronunciation of the word string W. With these chains of states, we can consider a set of state sequences $Q = (q_1, q_2, \ldots, q_m)$ in which at each time i, an observation from the hypothesis observation sequence A is assumed to be emitted by the corresponding state q_t in Q. Thus the total probability of observing the string W given the observation A is equal to the sum of the individual possible state sequences Q:

$$P(A|W) = \sum_{Q} P(A|Q, W)P(Q|W) = \sum_{Q}\prod_{i} p(q_i|q_{i-1})p(a_i|q_i) \tag{2.7}$$

The evaluation of the probability that the string W emits the observation sequence A is computed in terms of $p(q_i|q_{i-1})$, which denotes the state transition probabilities, and of $p(a_i|q_i)$, which denotes the probability that observation vector a_i is emitted by frame q_i. This term is referred to as the state emission probability. The state emission probabilities are normally modelled using a mixture of Gaussian densities for every state q_i:

$$p(a_i|q_i) = \sum_{k=1}^{K} p(k)N(a_i, \mu_k, C_k) \tag{2.8}$$

where k Gaussian densities are weighted by the mixture weights $p(k)$. In this equation μ_k and C_k denote the mean vector and the covariance matrix of the k^{th} Gaussian component of the mixture.

Given a certain collection of HMMs and a topology (*i.e.*, a definition of the possible paths in the HMM set) of these HMMS the problem estimating the parameters of the Gaussian distributions and the HMM models from training data can be solved using the Baum-Welch algorithm [16]. During recognition (also known as decoding), the Viterbi algorithm [77] is employed to evaluate the best scoring path from a hypothesis given the observed data.

2.2 Experimental environment

2.2.1 The CMU SPHINX-3 speech recognition system

The Carnegie Mellon University Sphinx-3 system is a frame-based, HMM-based, speech recognition system capable of handling large vocabularies. The word modeling is performed based on subword units (phone set), in terms of which all the words in the dictionary are transcribed. Each phonetic unit considered in its immediate context (which we will refer to as triphone) is modeled by 5-state left-to-right HMM model. To reduce the parameter estimation problem, data is shared across states of different triphones. These groups of HMM states sharing distributions between its member states are called senones [40].

The language model employed in our recognition experiments is a bigram-based language model developed using a training text corpus. These language models are smoothed using the Good-Turing discounting procedure [45].

The acoustic features used for recognition consist of 39-dimensional acoustic vectors derived every 10 ms spanning an analysis window of 20 ms The feature vector consists of the first 13 cepstral coefficients (including $c[0]$, the frame energy) and two blocks of 13-dimensional coefficients, one composed of the “delta” cepstral features (velocity) and the other composed of “delta-delta” cepstral features (acceleration). Cepstral mean normalization is always applied at the utterance level to remove statisti-

cal biases of the mean which might have been introduced by linear channel distortion [1].

2.2.2 The set of corpora employed

The effectiveness of the techniques proposed in this dissertation will be evaluated by performing recognition experiments on the SPHINX-3 platform. Several databases have been made available to the speech recognition research community over the years by the Linguistic Data Consortium of the University of Pennsylvania and the National Institute of Science and Technology (NIST) and conform a set of well known test corpora. We will be referring to two of these publicly available speech databases and one proprietary database provided by Telefónica de Investigación y Desarrollo in Madrid. We describe here the general characteristics and sizes of these corpora:

- **Resource Management RM1:** The Resource Management is a corpus of microphone quality, read utterances of a small vocabulary domain. In this thesis we use the training files in the RM1 database plus the development and test files in a configuration totaling 4800 files for training and 1600 files for testing uttered by a total of 160 speakers, with a total of 14968 tokens for evaluation. This is a small vocabulary set with a lexicon composed of 1125 words. We have reduced its bandwidth to 4 kHz and downsampled it to 8 kHz in order to make it compatible with the speech codec input signal specifications.

- **TIMIT:** The DARPA TIMIT Acoustic-phonetic continuous speech corpus is the result of a joint effort by MIT, Texas Instruments, DARPA and SRI. This database is a corpus of microphone quality, phonetically balanced, read utterances. It consists of 4620 utterances for training, 1680 utterances for testing (approximately 14500 tokens in the test set). It is conformed by read sentences uttered by 630 speakers of the main dialectic regions of the United States. It has the advantage of being phonetically labeled. Its lexicon is composed of 6229 words. As in the case of Resource management we have band-limited and downsampled this corpus.

- **TID (Telefónica) cellular corpus:** This corpus was recorded over the GSM cellular telephone network in Spain and represents an important effort in the research of cellular telephony speech recognition. It is a small vocabulary corpus of a numbers and quantities domain (approx. 75 words in lexicon), with approximately 6000 training utterances and 3000 testing utterances (with approximately 14500 tokens in the test) spoken by approximately 1700 speakers. It was originally recorded at 8000 kHz. It has been manually transcribed and annotated for acoustic environment, speaker's dialect and other conditions. It contains a broad sample of speaker and environmental conditions.

2.2.3 The statistical significance of different recognition results

We discuss here a set of procedures that determine the extent to which recognition results can considered to be significantly different from one another between different systems, algorithms and configurations.

A "simple method" was proposed by Gillick and Cox [31] to determine whether the outcomes of two recognition experiments differ significantly. This method is based on the assumption of the independence of the recognition errors, and has the advantage that confidence ranges can be established independently of the specific outcomes, depending instead only on the size of the testing corpora, the levels of confidence, and the baseline accuracy of the recognizer. Unfortunately, the bounds of the results established using this technique, are normally greater than what would be obtained by an outcome-dependent test [31] such as the Matched Pair method or McNemar's test. Table 2.1 column 1, shows the range in absolute percentual points within which the word error rate results can range around the baseline without being statistically different for the three levels of confidence depicted in the table, using the TID database and the simple method described in [31] (*i.e.*, assuming the independence of the errors). Column two shows equivalent results for the RM and TIMIT databases. What is noteworthy about the results displayed in column 1 is that they relate to the fact that the TID database has 13091 tokens in the test set, and the accuracy of the baseline system

is 6.9%. We can see that any result we want to compare to the baseline would have to be larger than 0.52%, 0.63% and 0.84% in absolute WER scores (for the three levels of confidence proposed, respectively) in order to be consider statistically different. Table 2.1, column 2, shows the same results for a database that conforms the TIMIT and Resource Management sizes and baseline accuracies applying this simple test: *i.e.*,considering approximately 16000 tokens in the test set, and approximately 10.0% WER in the baseline system. We can see that the corresponding bound are slightly larger than those for TID database: 0.55%, 0.65% and 0.9% for the three levels of confidence proposed.

Confidence	**%WER Range (TID)**	**% WER Range (TIMIT & RM)**
0.10	0.52%	0.55%
0.05	0.63%	0.65%
0.01	0.84%	0.9%

Table 2.1 WER ranges for the three levels of confidence specified using the TID database (left) and the TIMIT and Resource Management database (right) assuming independence of recognition errors.

As we have said, the test employed above makes the assumption that the recognition errors are independent from each other across systems. More statistical significance tests are not constrained by such an assumption, at the expense that every time two system outputs are to be compared, the test needs to be performed again. These results, however, yield bounds which are considerably smaller than those obtained using the simple test. In order to get a sense of the difference between the bounds obtained using the simple test and the bounds using the set of methods that do not assume independence we compared the six systems running a TID test evaluation with a different language weight for each configuration. Table 2.2 shows the recognition results for these configurations. As we can see, the WER obtained range from 6.9% (baseline) to 7.3%. According to the ranges established above for statistical significance of this database using the simple test, these results are not significantly different

among each other as their differences (which are always less than 0.4%) are smaller than the bounds established in Table 2.1. However, Table 2.3 shows the outcome of the set of tests (matched pair word error test, denoted MP; and McNemar sentence error test, denoted MN) implemented using NIST toolkit for statistical significance. Details on these techniques can be found in [31]. In each box, representing every possible pair of comparisons, the number between parentheses denotes the identity of the configuration that is deemed to be better according to the corresponding test (MP, and MN). We can see that the bounds for which statistical diferent results are established are actually smaller than the bounds established in Table 2.1. In brief, we can see that for the TID corpus when two results differ by 0.2% absolute (*e.g.*, differences between configurations 1 and 5 and between configurations 5 and 6), the tests determined once that they were identical and once that they were different. For all the instances when the results differed by 0.3% (configurations 2 and 6, 3 and 6, and 4 and 6), the tests determined that the results were statistically different. These results give us an idea that we can establish 0.3% as a range in which the results can be expected to be different by a rigorous set of statistical tests in the TID database. We can expect results to be not very different for the TIMIT and the RM databases given that we established by the simple approach that the ranges of confidence are very similar across these three databases.

Experiment	LW	**% WER**
Cfg. 1	**9.5**	6.9%
Cfg. 2	**9.0**	7.0%
Cfg. 3	**8.5**	7.0%
Cfg. 4	**8.0**	7.0%
Cfg. 5	**7.5**	7.1%
Cfg. 6	**7.0**	7.3%

Table 2.2 WER results for 6 different Language Weight values, using the TID corpus.

	Cfg. 1	**Cfg. 2**	**Cfg. 3**	**Cfg.4**	**Cfg. 5**	**Cfg. 6**
Cfg. 1		same	same	same	same	MP(1) MN(1)
Cfg. 2			same	same	same	MP(2) MN(2)
Cfg. 3				same	same	MP(3) MN(3)
Cfg. 4					same	MP(4) MN(4)
Cfg. 5						MP(5) MN(5)
Cfg. 6						

Table 2.3 Table of statistical significance results between systems from table 2.2. Tests shown are Matched Pairs (MP) and McNemar's test (MN), indicating in parenthesis which system is better.

2.3 Sources of degradation in a mobile network ASR environment

The degradation in the signal due to the speech codec processing amounts only to a fraction of the overall degradation in wireless channels compared to the effect of additive noise and bit errors in transmission. Bit rate and choice of speech coding scheme play a significant role in the effect of coding on recognition accuracy. Figure 2.1 shows the results of recognition experiments in which we can observe that accuracy decreases as bit rate decreases (the coding standards labeled in the horizontal axis are sorted by increasing bit rate). Reduction of the sensitivity of a recognition system to the overall effects of speech coding in a communication link will result in more robust systems whose performance is more independent of the type and bit rate of the codec used in a particular communication channel.

It can be observed from Figure 2.1 that under mismatched conditions (such as when the data used to test a system are coded but the data used to train it are not), a typical relative increase in the error rate of a system due to coding can be as high as 20 percent [22,54,41]. As the demand for mobile telephony increases further reductions in codec bit rate codecs are expected, the importance of developing bit-rate insensitive recognizers will increase.

Besides the detrimental effects due to coding, there is a significant increase in error rate due to errors in transmission. Table 2.4 presents experimental results obtained by Haavisto [33] in which he showed that other factors besides the coding scheme influence the degradation in recognition accuracy. Specifically, he worked with the three existing GSM coding standards (enhanced full rate, full rate coding and half rate coding) and landline conditions (labeled PSTN). He showed that the carrier-to-interference ratio (C/I) of the data affects accuracy considerably. To reduce this effect, Haavisto showed that one can recognize using acoustic models that were trained under similar codec and noise conditions, but the combination of these conditions results in a

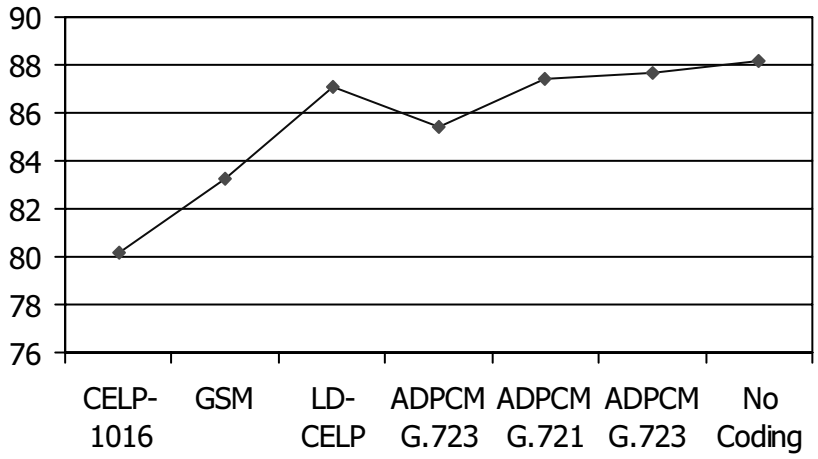

Figure 2.1 Effect of different speech codecs on speech recognition word accuracy (from Lilly and Paliwal).

large number of possible recognition scenarios. In normal operating conditions a classifier would need to determine how the signal was coded and the level of C/I present, having to add an initial classification stage to the system which naturally may be prone to errors. Finally, even if the identification of coding conditions is done successfully and models are available for every possible condition, we can see from Table 2.4 that the reduction in the recognition error introduced by these problems is partial: there is still a considerable large effect in recognition accuracy. Thus, it is necessary to develop solutions that raise the performance close to the non-coding levels (*i.e.*, the PSTN/PSTN performance)

,

Training/ Testing	PSTN	FR (error-free)	EFR (error-free)	FR (all)
PSTN	**3.6**	5.9	4.4	6.7
FR (error-free)	**9.0**	**4.8**	NA	5.1
HR (error-free)	**9.0**	NA	NA	NA
EFR (error-free)	**5.5**	NA	**4.3**	NA
FR (10 dB C/I)	11.1	5.3	NA	5.3
FR (7 dB C/I)	15.4	9.1	NA	7.0
FR (4 dB C/I)	50.0	**40.2**	NA	**21.2**
EFR (10 db C/I)	6.7	NA	4.8	NA
EFR (7 dB C/I)	9.9	NA	7.8	NA
EFR (4 dB C/I)	32.7	NA	32.7	NA

Table 2.4 Effect of the three GSM codecs at various bit error conditions on ASR WER under matched and mismatched conditions (from Haavisto [33]).

2.4 Recognition baselines

In this section we describe our baseline recognition experiments using the TIMIT database. We describe the configuration of our baseline recognizer, and present the results obtained under FR-GSM and CELP FS-1016 coding. We finally present the results we obtained using two off-the-shelf acoustic compensation techniques.

2.4.1 Baseline TIMIT experiments under GSM and FS-1016 coding

Our baseline TIMIT system consisted of a speaker independent, cross-word triph-one-based, continuous-density Gaussian mixture HMM recognition system modeled by approximately 600 senonically tied states [40], with diagonal covariance matrices. The language model in all of our experiments was a word bigram language model set with the language weight fixed to 9.5. This value provided reasonable speech recognition speed while keeping the contribution of the language model limited thus allowing the acoustic effects impact accuracy. In the later parts of this dissertation we have per-

formed experiments using several different number of Gaussian densities per mixture ranging between 8 and 64. As described in the previous section, our dictionary consisted of 6329 words.

Figure 2.2 shows the word error rate (defined as the number of correct words minus insertions, deletions and substitutions divided by the total number of words in the test set). Results are shown for matched conditions, in which the system is trained and tested using clean speech or the system is trained and tested using GSM-coded speech. Missmatched conditions correspond to the case where the testing data does not belong to the same environment in which the training data belongs. Missmatched conditions normally produce higher error rates than matched conditions. Results are also shown for the case where multistyle trainning conditions were applied. In multistyle training conditions utterances belonging to Clean and to GSM environments are employed. The results obtained using multistyle trainning and GSM coded test speech are very similar to the GSM matched conditions case. This suggests that that for the case of GSM coded speech, the benefit of combining GSM coded speech with clean speech does not result in significant advantages in recognition performance. We can see that the absolute difference between the Clean/Clean and GSM/GSM conditions is between 0.7 and 1.0 per cent, which corresponds to a 10% increase in word error rate for the case of 32 Gaussians per mixture.

Figure 2.3 shows the results of the experiments conducted using FS-1016 CELP coding (which we will refer to simply as CELP). In the experiments with the CELP codec a similar decoding configuration as in the GSM experiments was employed (*i.e.*, same dictionary, model definition, language model, etc.). In this case, the best CELP/CELP system is obtained using 32 densities per mixture, but very similar results are obtained with 16 densities per mixture. The degradation introduced by coding is between 2.0 and 2.5 percentual points in these two conditions, corresponding to a 25% increase in error rate between the best configuration for Clean/Clean and the best configuration in CELP/CELP cases. Results are also displayed for the CELP/Multistyle

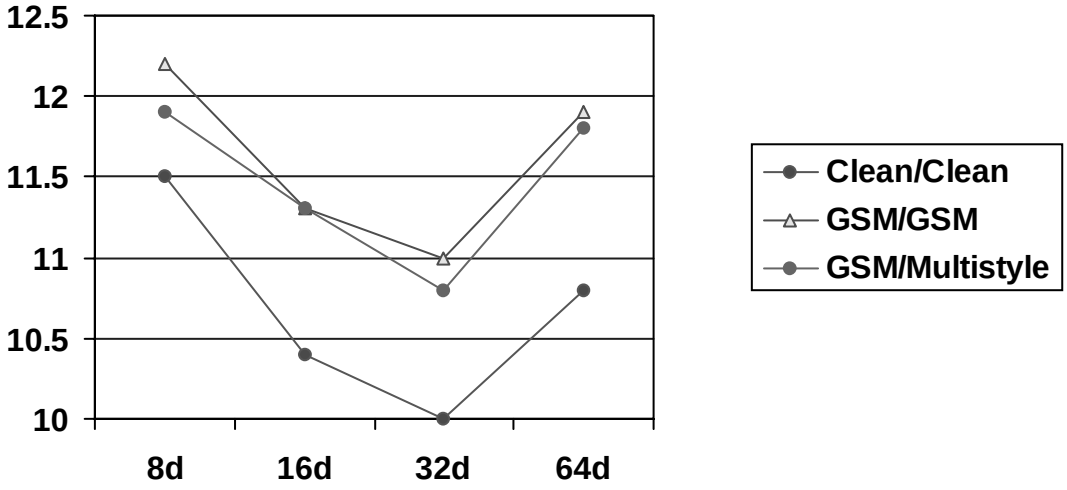

Figure 2.2 Baseline TIMIT recognition experiments WER with clean conditions and GSM conditions using different numbers of Gaussians per mixture.

configuration, in which, similarly as the GSM case, the results obtained are close to those obtained using matched conditions.

In the experiments described above best results were obtained using a configuration consisting of 32 Gaussian densities per state. When 64 Gaussians per density were employed the error rate increased indicating a data overfitting problem. In this thesis we will perform experiments using 8, 16 and 32 Gaussians per HMM state.

In all our baseline experiments as well as in all experiments in this thesis, cepstral mean normalization (CMN) has been consistently applied [25]. This simple procedure has been associated with the removal of linear time invariant filtering and due to its simplicity and benefits we include it as part of the baseline procedure.

2.4.2 The application of CDCN and MLLR compensation techniques

The table below shows the WER resulting from experiments performed using the codebook dependent cepstral normalization (CDCN) and the MLLR compensation techniques, with an 8 Gaussians per density configuration. The CDCN technique intro-

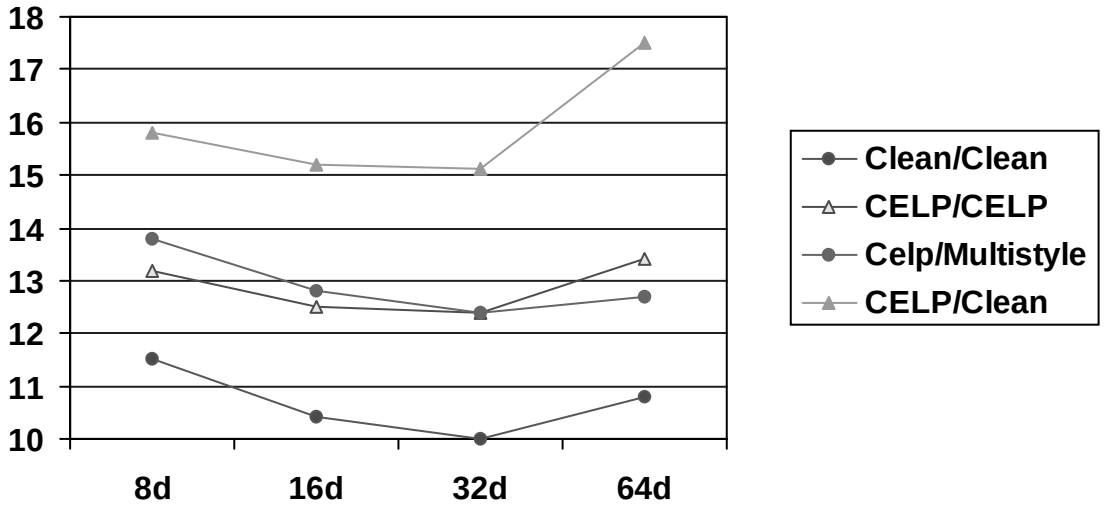

Figure 2.3 Baseline TIMIT recognition experiments WER with clean conditions and CELP conditions using different numbers of Gaussians per mixture.

duced by Acero [1] is intended to remove the effect of linear time invariant (LTI) filtering and stationary additive noise. It requires the use of a codebook describing the distribution of the clean data, and based on an estimate of the channel and the noise vector and on a mathematical model of the effect of this filtering and noise on the cepstral feature attempts to remove this effect. The channel bias and the noise are computed recursively, until an estimate of these vectors converges. We can see from Table 2.5 that the application of this technique under matched conditions (GSM/GSM) produces no improvement, resulting in exactly the same WER as the baseline conditions.

The same table shows the results for the recognition experiments with the Maximum Likelihood Linear Regression (MLLR) technique applied to mean adaptation. This technique was introduced by Leggetter and Woodland [53] and is a method for adapting the means of the distributions of the HMMs based on a multiplicative regression matrix and an additive bias vector. The estimation of the regression matrix and the additive bias is performed using the hypothesis of a first decoding pass and computing the matrix and vector that maximizes the likelihood of the hypothesis of this pass. Sev-

eral regression matrices can be obtained for different phonetic groups and the HMMs associated with these phonetic categories can then be compensated using the corresponding matrix and bias. This technique is called multiclass MLLR. MLLR does not makes no specific assumptions about the nature of the degradation, unlike other techniques based explicitly on additive noise, linear filtering, vocal tract length. From Table 2.5 we can see that for TIMIT, MLLR fails to produce positive results; the accuracy actually decreases by 0.9% in absolute terms.

Models/Compensation	% WER on GSM test speech
Clean/No Comp.	11.5%
GSM/No Comp.	12.2%
GSM/CDCN	12.2%
GSM/MLLR	13.1%

Table 2.5 Experiments using CDCN and MLLR compensation on GSM-coded TIMIT data.

We believe that the two off-the-shelf techniques tested in this section failed to produce positive results because several of their assumptions and requirements were not met. In the case of MLLR, a single regression matrix is not able to model the effect of the GSM coding on the means of the HMM distributions. While a larger number of regression matrices might provide better results, the amount of acoustic material needed to estimate these matrices was not available in this case.

For the case of CDCN, we believe that the existence of two conditions resulted in the lack of improvement. First, CDCN was designed to operate when recognition was based on clean mismatched models, and in favorable conditions this technique helps recognition get close to the matched model or retrained conditions. In our case, we observe that in *matched* conditions there is no gain in accuracy. Secondly, GSM coding

is not linear filtering or a source of simple stationary additive noise as CDCN assumes the effect of the environment to be.

2.5 Current work on mobile network and terminal ASR

Research into the problem of robustness in mobile speech recognition applications has addressed some specific issues such as the effect of coding and quantization on the recognizer and tandeming of such codecs. Salonidis and Digalakis [72] studied the application of Probabilistic Optimal Filtering to various coding topology scenarios. Lilli and Paliwal [54] studied the effects of coding and the relationship between bit-rate and tandeming on ASR performance. Euler and Zinke [22] also studied the effect of coding and recognition, and analyzed the performance of different cepstral features for recognition. Haavisto [33] presented a similar analysis but he included the effect of errors in the bit stream as well. Paping *et al.* [67] studied the detection of idiosyncrasies produced by the GSM network in order to recover or conceal transmission errors. Fissore *et al.* [24] also focused on the problem of hole detection, end-point detection and feature selection in the context of GSM speech recognition.

Other efforts (Gallardo-Antolin *et al.* [27], Huerta and Stern [41], Gallardo Antolin *et al.* [28]) have focussed on the assumption of the availability of the codec parameters during recognition and the use of such parameters for ASR feature derivation. Ramaswamy *et al.* [71] and Digalakis *et al.* [18], have studied the problem of quantization of the features prior to recognition. Their conclusion is that it is possible to design encoding and quantization algorithms that allow significant compression the speech features (or transformations of them) without affecting recognition performance. These compression techniques would be designed keeping the ASR performance in mind. Tucker *et al.* [75] report that perceptual quality and recognition performance are not incompatible, and they are able to demonstrate this using very low bit rate compression of the LPC information, ignoring the residual signal.

Gupta *et al.* [32] focussed on the robustness to environmental noise encountered in mobile applications using a two-level cepstral mean subtraction robustness approach. Soulas *et al.* [73] approached the problem focusing on adaptation of the PSTN models to a GSM environment using a Linear Multiple Regression Approach. Mokbel *et al.* [59] employed a wide set of classical approaches to robustness to the problem of GSM recognition. Specifically, they applied cepstral normalization, filtering of cepstral trajectories, blind equalization spectral subtraction and Bayesian HMM model adaptation.

Lawrence and Rahim [52] applied integrated bias removal techniques to the problem of mismatched speech recognition over a cellular network. Delphin-Poulat *et al.* [15] proposed a Multipath Stochastic Equalization framework to perform bias removal and linear regression, as well as equalization on the signal. In Dufor *et al.* [20] the use of a root-normalized front end as an alternative to MFCC features was proposed in the context of GSM recognition. Puel *et al.* [68] proposed the use of robust HMM architectures such as multi-HMM and multi-transition models, while Karray *et al.* [46] employed these methods as well as specific GSM hole rejection techniques for robust GSM recognition. Das *et al.* [12] proposed the use of MAP adaptation and LDA to achieve network independence in cellular environments.

There has been also a significant effort on the development and implementation of applications in mobile environments. Many of these efforts focus on the development and problems associated with in-car speech recognition and voice dialing.

Our research will focus the effect of speech coding on speech recognition. Our approach will specifically focus on analyzing the effect of speech coding techniques and identifying the source and nature of the distortion introduced to the signal by the codec; we then direct of effort on developing robust acoustic modeling methods based on the properties of the distortion introduced by the codec. In contrast to the approach and work done by others and which we described previously on this section, our work will focus on the analysis of the source of the degradation introduced by the codec.

2.6 Summary

In this chapter we introduced the problem of speech recognition when speech coding is present. We saw that speech coding can be a significant source of error during recognition, particularly when the acoustic models are mismatched. We then presented our recognition baselines and applied two off-the-shelf robustness techniques.

We saw that the MLLR and CDCN algorithms are unable to provide any reduction in the degradation introduced by coding. Even though there has been a wide interest and a significant effort in the area of speech recognition under coding distortion, most of these efforts focus on translating, applying and customizing well known and general adaptation and compensation techniques. In this thesis we will focus on robust acoustic modeling techniques based on an analysis of the operation of the codecs.

Chapter 3
Short-time autoregressive analysis for speech recognition and coding

Current speech codecs can be broadly classified in two categories: waveform-approximating codecs and parametric codecs [48]. Codecs within the waveform-approximating family are further subclassified into predictive and subband codecs. The predictive codecs first perform a short-term analysis of the signal in which the first autocorrelation coefficients of the speech signal, or short-term autocorrelation coefficients, are used to derive a linear prediction filter. Using the resulting linear prediction filter , the speech signal is filtered producing the short-term residual. The short-term residual is then coded using a long-term analysis procedure.

In essence, the goal of the procedure described above is to capture the short-term information which models the envelope or smooth part of the spectrum and separately code or represent the long-term information of the residual signal, which contains the fine detail part of the spectrum. Typically the short-term analysis is carried over frames of speech 20 to 25 ms long. The frame rate for the short-term analysis is typically 50 frames per second. Frames are usually segmented into subframes, typically 5 ms long. The long-term analysis is performed at the subframe level, across groups of subframes.

As mentioned in Chapter 2, speech recognition systems use mel cepstral features for recognition. This type of feature captures the broad spectral envelope of the speech signal. Cepstral features are usually obtained by transforming the smoothed and frequency-warped log power spectra of the signal. Cepstral features can also be derived from LPC coefficients which in turn are obtained by short-term analysis of the speech signal. As opposed to speech coding, speech recognition front-ends do not perform long-term analysis, or employ any information related to the residual signal.

This chapter first describes the basic idea behind the linear predictive analysis of speech and how this information is derived from the original speech signal. It then describes the form in which the residual signal gets encoded by long-term prediction analysis. We then describe the different ways in which the linear predictive information can be represented both for speech coding purposes and for speech recognition purposes. Finally, we describe the operation of the full rate GSM speech codec, and of the FS-1016 CELP codec that will later be employed in this thesis to test the techniques we propose based on the short-term/long-term framework that we establish.

3.1 Linear prediction based coding

Linear prediction based speech codecs assume that the speech signal is the output of an all-pole filter excited by an input signal (called the short-term residual, or excitation signal). This model has been associated with a simplified lossless tube model of piece wise constant area, which in turn approximates the behavior of human speech production [70]. Using z-transform notation we can express this model as the product of an excitation signal $E(z)$ and a shaping filter $\frac{1}{B(z)}$,

$$S(z) = \frac{E(z)}{B(z)} = \frac{E(z)}{1 - \sum_{i=1}^{M} b_i z^{-i}} \tag{3.1}$$

In the expression above, $S(z)$ is the z-transform of the speech signal, the numerator $E(z)$ represents the excitation, or residual signal which is associated with the glottal waveform, and the denominator $B(z)$ is the LPC polynomial that is viewed as a filter modeling the vocal tract and nasal cavity. $E(z)$ is the source which excites the LPC filter $\frac{1}{B(z)}$ to synthesize the speech signal. Therefore this is called the *source filter model*, or the *synthesis model*.

To separate the spectral envelope of the speech signal from the excitation component, speech codecs first perform an LPC analysis on the speech signal and obtain the LPC polynomial $B(z)$. Then, the excitation component is obtained by inverse filtering the speech signal:

$$E(z) = S(z)B(z) \tag{3.2}$$

This model is called is the analysis model [56, 70].

3.1.1 Short-term analysis: the LPC model

The denominator polynomial $B(z)$ in Equation 3.1 is defined by the predictor coefficient terms $b_1, \ldots, b_M$. This polynomial has M roots, referred to as the poles of the prediction filter. In this work we employ the sign convention in [70] for the signs of the coefficients of the predictor polynomial.

In the sampled data domain Equation 3.2 is equivalent to

$$e[n] = s[n] - \sum_{i=1}^{M} b_i s[n-i] \tag{3.3}$$

Defining $\hat{s}[n] = \sum_{i=1}^{M} b_i s[n-i]$ as the predicted sample sequence, the residual $e[n]$ can be interpreted as the difference between the actual sample $s[n]$ and the predicted sample $\hat{s}[n]$.

$$e[n] = s[n] - \hat{s}[n] \tag{3.4}$$

Hence $e[n]$ is also referred to as the prediction error sequence.

As mentioned above, $B(z)$ can be viewed as the filter through which the excitation or residual signal is driven in the synthesis model. The LPC filter is then an all-pole filter whose coefficients can be found by minimizing the mean squared error between the predicted and true value of $s[n]$:

$$E = \sum e^2[n] = \sum_n \left[s[n] - \sum_{k=1}^{M} b_k s[n-k] \right]^2 \tag{3.5}$$

To find the predictor coefficients that minimize the mean squared error, E is differentiated with respect to each b_k and each resulting expression is set equal to zero. The resulting set of equations when expressed in terms of the correlation function of the signal are a set of M linear equations that lead to the solution of the terms b_k.

3.1.2 Long-term analysis: the excitation model

The speech signal has, in its voiced segments, strong long-term correlation components due to the quasi-periodic vibration of the vocal chords [70]. While the broad spectral properties are modeled by the analysis filter, the long-term correlation information is retained in the residual signal $e[n]$. In order to be able to represent the excitation signal with a reduced number of bits, speech codecs remove the redundancy left in the residual signal by exploiting this long-term correlation. The residual in any subframe frequently resembles the residual in adjacent subframes. Based on this observation, speech codecs approximate the residual in any subframe according to its resemblance to the residual in previously reconstructed subframes.

A simple and good first approximation to the excitation of the current subframe, then, is the excitation signal of adjacent subframes. Speech codecs implement this

approximation by finding the gain and lag that minimize the difference between the reconstructed excitation signal in previous subframes and the excitation in the current subframe [51],

$$\frac{1}{P(z)} = \frac{1}{1 - \beta z^{-D}} \tag{3.6}$$

where β is the predictor coefficient or long-term gain and D is the predictor delay, or estimate of the pitch period. By performing the long-term prediction (LTP) of the excitation signal (or short-term residual), the predictable or periodic part of the residual is captured. Speech codecs utilize a second set of parameters (*e.g.*, multi-pulses, regular-pulses or fixed codebooks) to model the non-predictable or non-periodic portion of the excitation signal.

The weighted addition of a predictable component and an unpredictable component produces the reconstructed version of the excitation or reconstructed short-term residual. These two contributions can be interpreted to be based in two codebooks: one adaptive codebook, and one fixed codebook.

In the codebook-based interpretation of the residual coding, the adaptive codebook is generated at each subframe based on previous subframes of the reconstructed excitation signal. The index p from a codebook of M dimensional vectors denotes the best entry of the adaptive codebook; such entry is then weighted by $\lambda^{(a)}$. The index p is equivalent to the lag D in Equation 3.6 above and the gain $\lambda^{(a)}$ is equivalent to the gain β in the same Equation.

Similarly, the fixed codebook is assigned index q and gain $\lambda^{(f)}$. Expressing the overall reconstructed excitation signal in vector form in terms of the adaptive and fixed codebook entries ($c^{(a)}$ and $c^{(f)}$) and adaptive and fixed codebook gains ($\lambda^{(a)}$ and $\lambda^{(f)}$),

$$e_{pq} = \lambda^{(a)} c^{(a)}_p + \lambda^{(f)} c^{(f)}_q \quad (3.7)$$

The search for the optimal adaptive and fixed gains and codebook entries must be performed simultaneously. In order to simplify the search complexity, the adaptive gain and index are typically obtained first, and afterward the fixed codebook and index are determined.

3.2 Autoregressive representations of the speech signal

Both speech recognition and speech analysis aim to obtain an autoregressive or LPC-based representation of the speech signal with similar objectives. Speech coding tries to capture the short-term information of the signal leaving the problem of capturing long-term information of the excitation signal as a second analysis step. This short-term analysis information captures the shape of the spectral envelope of the speech signal which is associated with the filtering of the excitation produced by the vocal tract. In speech recognition, information related to the excitation signal (*i.e.*, the glottal waveform) is considered to be of reduced importance for recognition purposes: the linguistic message that the user is transmitting is represented by the vocal tract filter and the excitation signal is merely considered a carrier of this information. Speech recognition front ends perform either a short-term or reduced order LPC analysis, or they compute a transformation of the smoothed log-power spectrum of the signal. These operations attempt to retain as much short-term information as possible while trying to eliminate from the recognition features as much long-term information as possible. The following subsections describe the specific parameters usually derived from short-term autoregressive analysis of speech for speech coding and recognition, and the relations between these parameters.

3.2.1 Autoregressive representations for speech coding

The coefficients b_k in Equations 3.1 and 3.3 above constitute the prediction polynomial and are sufficient to represent the LPC spectral information. However, because speech coding algorithms impose significant quantization noise on the parameters to transmit, typically the LPC information is transformed to representations which show more robust distortion behavior under speech coding (*e.g.*, more stable, less sensitive to quantization etc.)

- **Reflection coefficients (or PARCOR coefficients):** The reflection coefficients K_i can be derived from the LPC coefficients and from the autocorrelation coefficients [56, 70]. For speech coding, these coefficients present two advantages over their LPC counterparts [65]: numerical stability is guaranteed if the coefficient values are kept within the range -1 to +1, and they are less sensitive than LPC coefficients to spectral distortion introduced through quantization. The reflection coefficients then can be recursively derived from the following expressions:

$$
\begin{aligned}
E^{(0)} &= r(0) \\
k_i &= \frac{\left\{ r(i) - \sum_{j=1}^{L-1} \alpha_j^{(i-1)} r(|i-j|) \right\}}{E^{(i-1)}}, \quad 1 < i \le p \\
b_i^{(i)} &= K_i \\
\alpha_j^{(i)} &= \alpha_j^{(i-1)} - K_i \alpha_{j-j}^{(i-1)} \\
E^{(i)} &= (1 - K_i^2) E^{(i-1)}
\end{aligned}
\tag{3.8}
$$

where K_i are the reflection coefficients.

- **Log area ratio coefficients**: The LAR coefficients L_i are a simple linear transformation of the reflection coefficients and are employed in order to overcome the reflection coefficients drawback of being more sensitive to quantization when they represent narrow-bandwidth poles (*i.e.*, when the reflection coefficient's magnitude is close to 1)[69]. Their relation to reflection coefficients is:

$$L_i = \log\left(\frac{1 + K_i}{1 - K_i}\right) \tag{3.9}$$

- **Line spectrum frequency representation**: The LSF were introduced by Itakura [43], Among the properties of this representation are a bounded range with check for stability and sequential ordering. Due to being a frequency representation can be associated with techniques that exploit human perceptual properties [65]. The LSF are the roots of the polynomials $P(z)$ and $Q(z)$ that are constructed in terms of the LPC vector $B(z)$ using the following equations:

$$\begin{aligned} P(z) &= B(z) + z^{-(M+1)}B(z^{-1}) \\ Q(z) &= B(z) - z^{-(M+1)}B(z^{-1}) \end{aligned} \tag{3.10}$$

3.2.2 Autoregressive representations for speech recognition

Most speech recognition systems employ cepstral features of limited order (typically 13) because they are a more robust and reliable feature set than LPC, reflection coefficients or LAR coefficients for ASR (*cfr.* [69]). Because of the existence of efficient FFT routines, mel frequency cepstral coefficients are most frequently derived from the smoothed, and warped short-time spectrum of the speech (see section 2.1.1). Even though the use of the LPC coefficients as a feature for speech recognition is atypical, it is possible to derive cepstral coefficients from them. In later chapters we will be making use of LPC-Cepstra in experiments deriving cepstra from codec information.

We describe here how the LPC-cepstra is derived and we also describe briefly the PLP front end that is based on an autoregressive analysis.

- **LPC Cepstra:** The cepstral feature vector can be obtained from the LPC vector through the following recursion [69]:

$$c_m = b_m + \sum_{k=1}^{m-1} \left(\frac{k}{m}\right) c_k b_{m-k} \tag{3.11}$$

where c_0 is the log energy of the frame, and $1 \le m \le M$. That is, the above expression is true for the first M cepstral coefficients where M is the LPC order. For $m > M$:

$$c_m = \sum_{k=1}^{m-1} \left(\frac{k}{m}\right) c_k b_{m-k} \tag{3.12}$$

- **Perceptual Linear Prediction**: Hermansky [36] proposed a class of features based on linear predictive analysis techniques that modify the power spectrum of speech by applying critical-band spectral resolution, equal-loudness curve, and intensity-loudness power law. Cepstral features are then derived from this perceptually LPC representation of the speech signal for automatic speech recognition using the relations described in Equation 3.12 above.

3.3 The Full Rate GSM speech codec

The full-rate GSM codec is a linear predictive regular-pulse excited-long-term predictive (RPE-LTP) based codec operating with a bit rate of 13 kbps [23]. The 8-kHz speech signals enter the codec where they are analyzed in frames of 160 samples from which the 8^{th}-order LPC parameters are obtained every 20 ms, thus producing an LPC

analysis rate of 50 frames per second. The LPC parameters are represented as log area ratio (LAR) coefficients which are quantized and then transmitted. Each set of LAR coefficients is represented using 36 bits, thus 1800 bits per second are allocated in LPC information (15% of the total bit rate).

The residual signal from the LPC analysis (*i.e.* the short-term residual) is subdivided into subframes of 40 samples each and coded by a regular pulse excited-long-term prediction codec whose quantized parameters are transmitted using the remaining 85% of the bits. In this section we detail the operation of the RPE-LTP coding process of the short-term residual [50, 76] in the GSM full rate codec. The work in this dissertation was based on the publicly-available implementation of the GSM codec in C by Degener and Bormann [14].

The RPE-LTP codec can be described in simplified form as a two-part process: a long-term predictor process (the LTP block) that produces an estimate of the short-term residual signal, and a Regular Pulse Excitation process (the RPE block) which is responsible for representing the “unpredicted” part of the short-term residual signal (called the long-term residual signal) using a reduced number of bits. Under normal conditions, the LTP block will try to capture the long-term periodicity of the signal associated principally with voiced speech segments based on a subframe cross-correlation analysis. We now explain these concepts in more detail.

For the purpose of illustration we present two diagrams representing simplified versions of the RPE-LTP codec that process the short-term residual signal that comes out of the LPC analysis. The two diagrams we present correspond to two versions of the RPE-LTP codec: an ideal codec and a real codec. By comparing and contrasting these simplified codecs we can identify the source and nature of the distortion introduced to the reconstructed version of the residual signal. In the following chapter we will relate the behavior of the signal distortion to degradation in recognition accuracy.

Figure 3.1 is a simplified block diagram of an ideal RPE-LTP codec. The primary difference between the ideal codec and a real RPE-LTP codec is that the ideal codec does not produce quantized versions of its signals or parameters. For this reason, the ideal codec does not achieve any reduction in bit rate. The short-term residual signal $e[n]$ enters the ideal codec and is compared to the short-term residual estimate $\bar{e}[n]$ produced by the LTP block. The difference between these two signals corresponds to the part of the residual signal which the long-term Predictor block was unable to predict. This signal is called the long-term residual signal $r[n]$, and it represents what needs to be added to the short-term residual estimate to obtain the reconstructed short-term residual signal. In other words, this signal represents a sort of “innovation” or unpredictable part of the short-term residual signal. The decoder section of the codec contains an identical long-term Predictor block which generates a short-term residual estimate, based on the received LTP parameters and the previously reconstructed version of the short-term residual. After the short-term residual estimate is generated, the ideal codec adds the received innovation part of the signal (*i.e.*, the long-term residual) to it. Because the sum of the long-term residual and the short-term residual estimate signal results in exactly the residual sequence, the ideal codec produces no loss or distortion in the restored signal: in this ideal case the reconstructed short-term residual $\hat{e}[n]$ and the short-term residual $e[n]$ are equal. However, the ideal codec must trans-

mit an exact copy of the long-term residual signal to achieve this, so its bit rate is no less than the bit rate of the original short-term residual sequence.

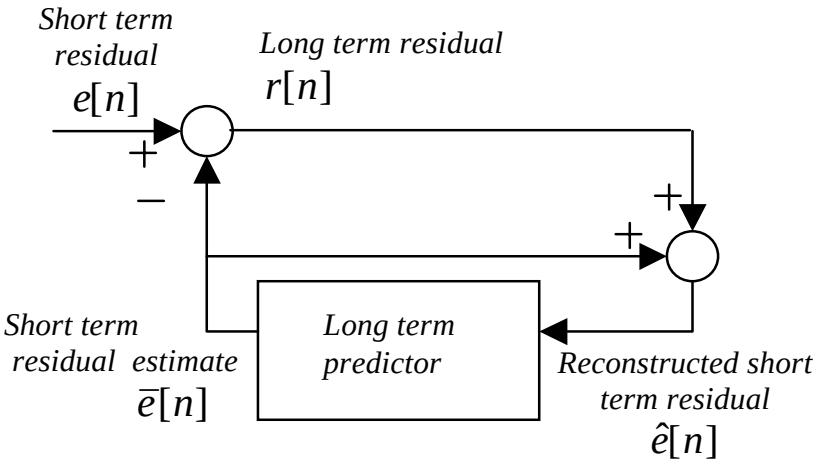

Figure 3.1 A simplified block diagram of an ideal RPE-LTP short-term residual codec.

In reality, the RPE-LTP coder transmits a subsampled and quantized approximation of the long-term residual sequence and the LTP information in order to achieve bit-rate reduction. Generally, the coder does not provide all the information that is needed to obtain a perfect reconstruction. The reconstructed representation of the long-term residual obtained from the transmitted information (called the quantized long-term residual $\hat{r}[n]$, or reconstructed long-term residual) is only an approximation to the original innovation sequence. Figure 3.2 illustrates this process by adding to the codec the block labeled *RPE coding.* The amount of degradation in the reconstructed signal will be related to the energy of the original long-term residual signal which in turn depends on how well the long-term predictor module in the coder is able to "follow" or predict the next subframe of the time sequence based on previous reconstructed subframes.

The RPE codec introduces distortion to the quantized long-term residual that is proportional to the energy present in it. From the analysis of the operation of the RPE-LTP codec above, we suggest that the energy of the long-term residual can be associated with the predictability of the short-term residual. Because the different phones of

any given language can be associated with a certain level of periodicity, or predictability (for example, vowels are likely to be more predictable than consonants), we can expect to find certain patterns in the distribution of the amount of distortion introduced by the RPE-LTP coding process. We will illustrate this point in the next chapter.

Other existing coding schemes in which the error minimization block consists of a predictive component (*i.e.*, closed-loop prediction-based coders, [51] [48]) can be thought to operate in a similar fashion as the basic system of Figure 2, with the main differences between codecs being the way the long-term prediction is performed and how the long-term residual gets represented and the effects of this quantized representation in the reconstructed long-term residual.

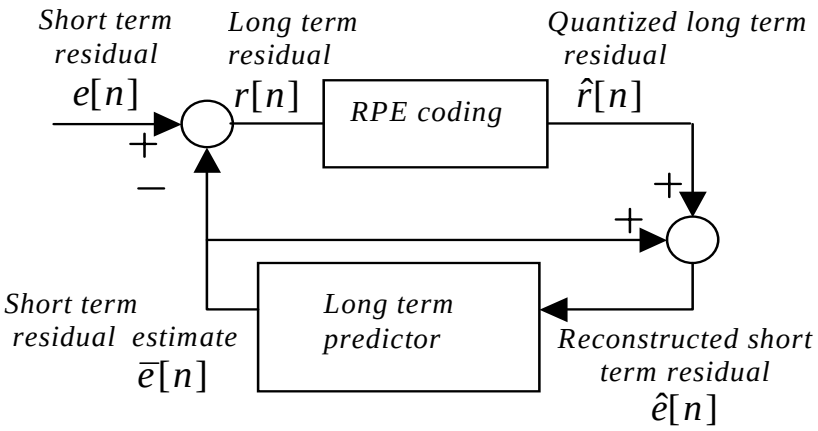

Figure 3.2 A simplified block diagram of a real RPE-LTP short-term residual codec.

3.4 The FS-1016 CELP speech codec

The FS-1016 speech codec [10] is the result of a U.S. DoD program launched to develop a third-generation secure telephone unit by the use of 4.8 kbps modem technology. In 1988, the DoD conducted a survey of 4.8 kbps speech codecs to be used in such

secure systems [47]. The selected codec was developed by the DoD and AT&T Bell labs.

The FS-1016 standard is based on an 10^{th} order LPC analysis, followed by a CELP representation of the short-term residual signal. The long-term periodicity is modeled by an adaptive codebook (which is an equivalent process to the LTP block of the GSM codec). The adaptive codebook is generated from previous subframes of the reconstructed short-term residual. The long-term residual signal, or difference between the short-term residual and the short-term residual signal estimate, is coded by means of a fixed ternary stochastic codebook. The use of ternary values in the stochastic codebook (-1,0,+1) allows for fast search of the optimal codes. The selection of the optimal scaled excitation vectors (both adaptive and stochastic) is performed by minimizing a time varying, perceptually weighted distortion measure. This codec uses, as does the FR-GSM counterpart, 8 kHz as sampling rate. Its frame size is 30 ms long with four 7.5 ms subframes per frame.

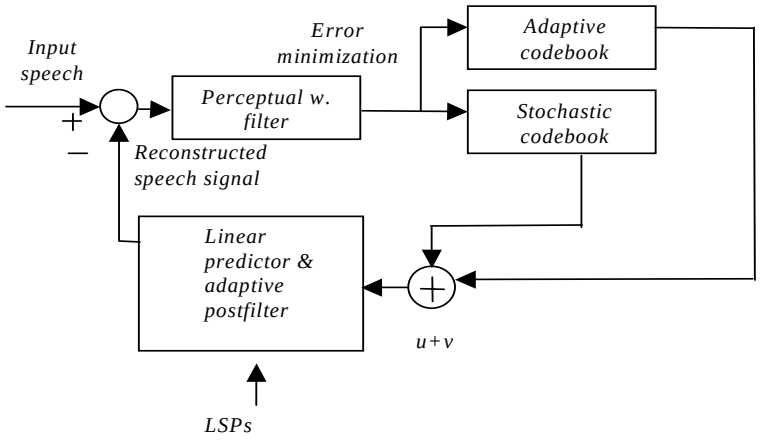

Figure 3.3 Simplified block diagram of the CELP block of the FS-1016 codec.

Figure 3.3 shows a simplified block diagram of the operation of the overall FS-1016 analyzer. The major similarities with the GSM codec are the closed loop prediction, and the decomposition of the analysis of the short-term residual into a fixed stochastic codebook component (corresponding to the RPE analysis of GSM) and an

adaptive codebook component (corresponding to the LTP block in GSM) aimed at modeling the pitch. In the next section we compare the GSM and FS-1016 codecs and discuss how these differences might affect ASR.

3.5 Comparison of the FR-GSM and FS-1016 codecs

In subsequent chapters we present recognition experiments performed using the FR-GSM codec. Some of the techniques we propose were designed taking into consideration the general closed-loop long-term prediction performed by the GSM codec. In order to evaluate the generalizability of our algorithms to other codecs that work in similar ways (*i.e.*, closed loop long-term prediction), some of our techniques will be applied to speech coded with the FS-1016 codec.

In some respects the FR-GSM codec presents properties that seem, at least intuitively, more benign for ASR performance. However, in this section, we will perform a detailed comparison between the properties of both standards and observe that this is not necessarily the case for every single property. Empirically, we have observed in other parts of this dissertation that the FS-1016 codec affects recognition in a larger extent than the GSM codec.

- **Bit Rate:** 4.8 kbps (FS-1016) *vs.* 13 kbps (GSM). In Chapter 2 we saw that the lower the bit rate, the larger the impact on recognition. FS-1016 compresses speech to less than half the bits employed by GSM coding.

- **LPC analysis order:** 10^{th} order (FS-1016) *vs.* 8^{th} order (GSM). The quantization process of the LPC parameters introduces less degradation in ASR performance than the residual or excitation coding; thus, recognition from FS-1016 speech seems to be more robust than from GSM speech in this respect.

- **LPC representation:** LSP (FS-1016) *vs.* LAR (GSM). LSPs are a representation that requires a larger amount in computation increasing the computational complexity of the codec, but this representation is widely used due to its greater robustness to spectral distortion due to quantization.

- **Frame/subframe rate:** 30 ms./frame (FS-1016) *vs.* 20 ms./frame (GSM). Both standards compute four subframes per frame. The FS-1016 frame is longer, so the codec algorithmic delay is potentially larger, but this is of no importance to ASR.

- **Quantization and representation of the long-term residual:** The GSM codec computes a LTP gain and lag, while the FS-1016 performs a joint search of the adaptive codebook and the fixed stochastic codebook.

- **Representation of the long-term prediction information:** The FS-1016 codec, which is based on CELP coding, uses a fixed stochastic codebook *vs.* regular pulses employed by GSM. CELP performs a vector quantization of the long-term residual, while the RPE representation of the long-term residual is equivalent to a scalar-quantization an subsampling of the original long-term residual. Quantizing a vector, instead of its scalar components results in less distortion for a given bit compression rate; in other words, for a given distortion it is possible to compress the signal with less bits.

- **Error metric/distortion metric employed:** The adaptive codebook and stochastic codebook search in the FS-1016 codec are performed using a modified minimum squared prediction criteria (MSPE) of the perceptually weighted error signal and the MSPE of the perceptually weighted error signal respectively [10]. The search for optimal LTP and RPE parameters is similarity performed using a minimum squared error criteria of the weighted error. The minimization of the energy of the error will minimize the SNR, however, it is not clear what is the effect of the use of a perceptual weight before the MSE computation on ASR accuracy.

- **Domain of error minimization:** Squared errors are minimized for GSM for the long-term residual, while for FS-1016 they are minimized for the reconstructed speech signal. In the case of FS-1016, the minimization of the error is performed on the signal that the user will actually have access to. When the minimization of the error takes place in the same domain the user will encounter, the overall effects of quantization are guaranteed to be taken into consideration: *i.e.*, the quantization introduced by LPC coding will be considered for CELP coding, while in GSM this distortion in not taken into account.

Overall the FS-1016 algorithm is a more complex coding scheme. These advantages would imply lower coding distortion and lower impact on ASR performance if the bit rates of both codecs were similar. The greater impact of the FS-1016 codec on recognition observed in the baseline experiments in Chapter 2 implies that the advantages of more sophisticated coding methods are lost because of the considerably smaller bit rate that this codec is producing.

3.6 Summary

In this chapter we described some basic ideas behind speech coding: the analysis of short-term and long-term information in the speech signal. Unlike speech coding, speech recognition typically employs only short-term information as feature. We further described the operation of the GSM and CELP codec and contrasted their differences. We concluded that the FS-1016 codec affects ASR to a greater extent than GSM because of its significantly lower bit rate, in spite of FS-1016 being more computationally complex and having better quantization properties.

Chapter 4
Speech recognition from GSM codec parameters

The cepstral features employed during recognition are a representation of the short-term information of the speech signal which differs from the short-term representation computed by the LPC analysis block of the GSM codec. In this chapter, our goal is to establish a relation between the cepstral feature used for recognition, and the short-term analysis and parametrization of the signal obtained in the GSM codec. We will explore the idea of transferring the speech parameters from one representation into the other without having to reconstruct the speech signal. Our main goal will be to circumvent the effect of coding and quantization introduced to the reconstructed long-term-residual by the RPE-LTP codec.

In order to accomplish this, we first analyze the nature of the distortion introduced to the cepstral representation of each of the two parameter streams in the GSM codec: the short-term and the long-term parameter streams. This analysis will be performed by extending the description of the functionality of the GSM codec that we introduced in the previous chapter into the cepstral domain. We also illustrate the effect of GSM coding in terms of its the impact on the means of the models of the HMMs. This effect of GSM coding on the cepstral features and HMM models does not have a closed analytic solution, so the approach proposed in this chapter does not attempt to produce a rigorous model of this degradation. Instead, we will try to circumvent the problem by mixing the information necessary for recognition in a way that takes into consideration the patterns of distortion introduced to these two parameter streams. This method effectively builds a connection between the cepstral representation of speech used in ASR and the parameters employed by the GSM codec.

4.1 The effect of FR-GSM quantization and coding on codec parameters and ASR cepstral features

In Section 3.3 we presented a description of the operation of the main blocks that constitute the full-rate GSM codec. We described the effect of the RPE codec's sub-sampling and coding on the long-term residual. In the following subsection we extend this analysis and perform some simulations to study the effect of coding and quantization on the codec parameters and its eventual effect on likelihood surfaces of the parameters used for recognition.

4.1.1 Mathematical description of the FR-GSM codec

In the analysis presented here we refer to the terminology developed in Sections 3.3 and 3.1.1, as well as in Figure 3.2. Since all the signals are with reference to the current subframe m, the notation denoting this subframe has been removed.

When the speech signal $s[n]$ enters the GSM codec, an LPC analysis pass is performed to obtain the log area ratio coefficients (LAR), L_i. These coefficients are quantized producing the set of quantized log area ratios (Q-LAR), $\hat{L}_i$. The speech signal is filtered through the LPC analysis filter derived from the Q-LAR (which we call the Q-LPC analysis filter), producing the short-term residual signal or prediction error $e[n]$. After this LPC analysis process, the undistorted original speech signal can be reconstructed through the convolution of the short-term residual and the impulse response of the Q-LPC analysis filter $\hat{h}[n]$,

$$s[n] = \hat{h}[n]*e[n] \tag{4.1}$$

By definition, the long-term residual signal $r[n]$ is equal to the difference between the short-term residual signal $e[n]$ and the short-term residual estimate produced by the LTP block $\bar{e}[n]$. Thus, the short-term residual signal can be expressed as,

$$e[n] = \bar{e}[n] + r[n] \tag{4.2}$$

The long-term residual signal $r[n]$ is coded and quantized by the RPE block. The quantized representation of this signal is called the reconstructed long-term residual $\hat{r}[n]$. The long-term prediction quantization error is then defined as the difference between the input to the LTP block and its output.

The quantization introduced by the RPE block is determined completely by its current input subframe $r[n]$. This quantization process is not a linear time-invariant system. As is the case with many compression processes, it can considered to be a many-to-one mapping; we refer to the quantization error introduced by the RPE block as function Υ which depends solely on $r[n]$,

$$\begin{aligned} q[n] &= \hat{r}[n] - r[n] \\ &= \Upsilon(r[n]) \end{aligned} \tag{4.3}$$

where $q[n]$ is the RPE quantization distortion signal.

Because in the RPE process some samples of the input are selected to be represented by regular pulses and the remaining samples are dropped, the energy of the RPE quantization distortion signal $q[n]$ is proportional to the energy of the input long-term residual signal.

The reconstructed short-term-residual signal $\hat{e}[n]$ is then, by definition, the addition of the short-term residual estimate $\bar{e}[n]$ and the reconstructed long-term residual, which can be expressed in terms of the original long-term residual signal,

$$\begin{aligned} \hat{e}[n] &= \bar{e}[n] + \hat{r}[n] \\ &= \bar{e}[n] + r[n] + \Upsilon(r[n]) \\ &= e[n] + \Upsilon(r[n]) \end{aligned} \tag{4.4}$$

We can see from above that the difference between the reconstructed short-term residual and the short-term residual is the long-term residual quantization term. The reconstructed-short-term residual is then filtered through the synthesis filter derived from the Q-LAR coefficients. Therefore, the resulting overall speech signal $\hat{s}[n]$ is,

$$\begin{aligned} \hat{s}[n] &= \hat{h}[n]*\hat{e}[n] \\ &= \hat{h}[n]*(e[n] + \Upsilon(r[n])) \\ &= s[n] + \hat{h}[n]*\Upsilon(r[n]) \end{aligned} \tag{4.5}$$

The expression above reflects the fact that the overall reconstructed speech after GSM coding is equal to the original speech signal plus an additive distortion quantization $d[n] = \hat{h}[n]*\Upsilon(r[n])$ term. This additive quantization term is the result of the convolution of the impulse response of the Quantized LPC synthesis filter Q-LPC with the RPE quantization term $\Upsilon(r[n])$, which in turn depends on the long-term prediction signal $r[n]$.

The effect of coding speech in the presence of additive environmental noise can be described similarly. We outline this analysis in Appendix A.

It is evident from Equation 4.5 that if we were to describe the effect of GSM coding on the speech signal as an additive noise, then the estimation of the additive component would be a difficult process, as this component is non-stationary, changing considerably from subframe to subframe, since it is a function of the unpredictable part of the excitation signal. This is a reasonable result given that the task of the speech codec is to remove long-term and short-term redundancy from the speech signal by exploiting long-term and short-term correlations. It is thus expected that techniques such as VTS and CDCN [1, 60] and other robustness methods based on the concept of a stationary additive interfering noise would be of limited benefit when trying to remove the effect of quantization.

4.1.2 Effect of GSM coding on the recognizer feature vector

The speech recognizer operates using cepstral features derived from the speech signal. In the previous subsection we have analyzed the effect of FR-GSM coding on the time-domain signal. In this subsection we extend the analysis of FR-GSM coding on the speech signal in the cepstral domain.

Acero, Moreno [1, 60] and others have developed robustness techniques based on estimates of the additive noise and a linear-channel model of the degradation. Their technique, however, assumes stationarity of the noise signal and linearity of the channel. Similarly, Lawrence and Rahim [52] proposed a technique for cepstral bias removal in which the assumption referring to the stationarity of the perturbation is waived but its effect on the cepstra is assume to be additive. A more accurate model of the GSM coding degradation would have no assumptions regarding stationarity, and furthermore would not assume to be additive in the cepstral domain. Due to the fact that the perturbation due to GSM coding varies substantially from subframe to subframe, and that it is related to the LTR (from which most of the long-term and short-term periodicity has been removed) it is very difficult to derive its estimates.

We now extend the analysis of subsection 4.1.1 to the cepstral domain. The reconstructed speech signal differs from the original signal by a non-stationary distortion signal $d[n]$, which has been defined as:

$$d[n] = \hat{h}[n] * \Upsilon(r[n]) \tag{4.6}$$

Based on the model of the effect of additive noise signals to the cepstral recognition feature (*cfr.* [1]), we can now relate the perturbing additive signal in this model to the perturbing signal $d[n]$. Let $\mathbf{s}$ denote the cepstral vector of the original speech signal, $\hat{\mathbf{s}}$ denote the cepstra of the reconstructed speech signal, $\mathbf{h}$ the cepstra of the LPC synthesis filter of the corresponding speech frame and Υ the cepstra of the noise due to RPE quantization for the same speech frame. We can express the resulting cepstra of the reconstructed speech signal in terms of the quantized LPC filter and the RPE quantization distortion,

$$\hat{\mathbf{s}} = \mathbf{s} + IDFT\left(\ln\left(1 + e^{DFT\left(\hat{\Upsilon}\hat{\mathbf{h}} - \mathbf{s} \right)} \right) \right) \tag{4.7}$$

where,

$$\begin{aligned} \mathbf{s} &= IDFT\{\ln S(w_i)\} \\ \mathbf{h} &= IDFT\{\ln H(w_i)\} \\ \Upsilon &= IDFT\{\ln \Upsilon(w_i)\} \end{aligned} \tag{4.8}$$

In these expressions $S(w_i)$ represents the power spectrum of the i^{th} window of the speech signal $s[n]$. Similar relations exist for the other signals: $H(w_i)$ represents the power spectrum of $h[n]$ and $\Upsilon(w_i)$ represents the power spectrum of $\Upsilon(r[n])$.

Evidently, the relation between the original cepstrum of the speech signal **s**, the cepstrum of the noise term Υ and the cepstrum of the resulting observed speech signal $\hat{\mathbf{s}}$ is not linear. No closed solution exists to find **s** given the observed data $\hat{\mathbf{s}}$ and estimates of the noise cepstrum Υ. For the stationary-noise and linear-channel case, the CDCN and VTS methods [1, 60] solve this estimation problem recursively. As mentioned earlier, the particular case of GSM coding is compounded by the degree of non-stationarity of the perturbation signal.

4.1.3 Effect of coding and quantization on ASR likelihood surfaces

We have briefly described in Chapter 2 the HMM based approach to speech recognition. We saw that the modeling is performed based on the pronunciations of words and that a language model component carries some form of grammatical and syntactical information. However, at the most basic level, speech recognition is a basic pattern classification problem in which the classes' decision surfaces are described in terms of Gaussian mixtures. In this section we illustrate the effect of GSM coding on this pattern classification process.

Figure 4.1 below shows the contour lines that describe the distribution of the first two cepstral coefficients (the horizontal axis represents the first cepstral coefficient and vertical axis the second) of the cepstral feature vector when two Gaussian densities per mixture are used to model the data. We use this representation to depict the models of two classes of data (phone **m** and phone **n**) with and without GSM coding. The top left panel corresponds to the density of Class 1 (that represents the first state of the context independent HMM of the phone **m**) with no GSM affecting the speech signal. The bottom left panel corresponds to a similar condition, except that it corresponds to Class 2 (that represents the first state of the context independent HMM of the phone **n**). The top right panel and bottom right panel correspond to the density surfaces of Classes 1 and 2, respectively, when speech is coded through GSM.

After observing the two panels in the top row we can notice that even though there are no radical changes in the general shape of the surface, one of the means of the mixture seems to be moving away from the other mean in the distribution. This makes the density seem to be more spread even though the variances of both Gaussian densities did not increase. Indeed, the variances of the both mixtures decreased. This phenomenon, observed previously by Moreno [60] in the case of stationary additive noise, is related principally to the relative magnitude of the variance of the distribution of the noise with respect to that of the SNR. For the particular case illustrated here the reduction in the variances of these Gaussians was small. Having a classification problem where classes have distributions that spread more in the feature space increases the probability of missclassification increasing the error probability area of the classifier [19].

We can see that the bottom left and right panels corresponding to Class 2 without and with GSM respectively have the same problem: the surfaces change subtly, not by substantially increasing the variances but mostly by modifying the mean locations. This distribution also looks more spread. We can see that the missclassification error area between Class 1 and Class 2 when there is GSM is larger than when there is not GSM present, thus making classification more difficult

In the previous section we mentioned that tracking the effect of the GSM coding on the cepstra of the speech signal is a prohibitive task given the non-stationary nature of the perturbation. In this subsection, we have described the average effect of this perturbation on the Gaussian distributions. For the classes illustrated here, the effect we observed accounts for an increase of the probability of error of the classifier by spreading and displacing the means of the Gaussians.

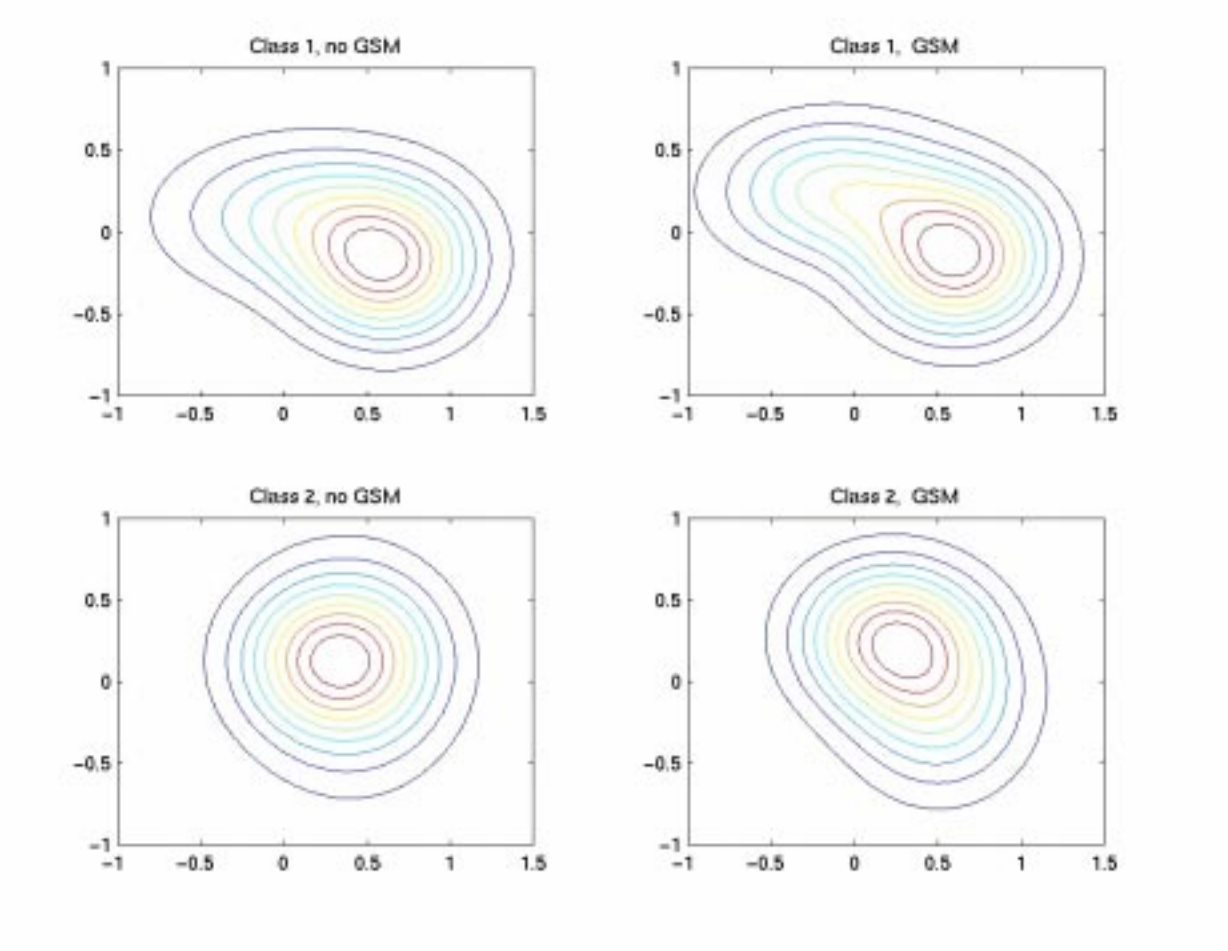

Figure 4.1 Contour surfaces of the first two cepstral coefficients of the feature vector when two Gaussian densities per mixture are used. The top left panel corresponds to the density of Class 1 (phone **m**) with no GSM. The bottom left panel belongs to Class 2 (phone **n**). The top right panel and bottom right panel, are the surfaces for Classes 1 and 2 with GSM.

4.2 Derivation of ASR cepstral features from GSM codec parameters

In Section 4.1.2 we described the effect of the GSM coding and quantization process on the cepstral features used for recognition. By doing this, we have implicitly assumed that the GSM coding process behaves like a “black box” that produces a single output signal based on a single input signal. We know however that within the codec there are two principal coding processes: the process associated with short-term

predictive analysis (LPC) and the process associated with the long-term predictive analysis. In this section we examine the level of recognition accuracy that can be obtained when going directly from one *compressed* representation of the speech signal (namely, he long-term/short-term information) into another compressed representation of the speech signal (namely, the cepstra).

The motivation for avoiding deriving cepstra from the reconstructed speech signal is twofold: first we can get an idea of the role or contribution of each parameter stream to recognition, and second we can possibly identify, characterize and reduce the effect of coding and quantization on the parameter streams. It is important to clarify here that the GSM codec *first* represents the speech signal in a short-term/long-term way *and then* it performs the actual bit rate compression by means of quantization and subsampling, therefore the parametrization and the coding processes are independent. In this section we describe how to derive cepstra from these two types of information and later on we will discuss how they can be recombined. We will also describe the importance of the two streams for recognition and the effect of compression and quantization on them.

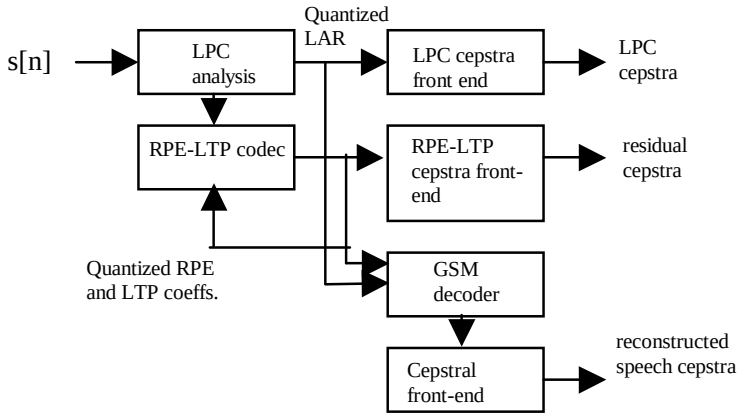

Figure 4.2 Diagram depicting the three possible types of sources of cepstral features at different stages of GSM coding.

4.2.1 Deriving cepstra for the LPC Log Area Ratio parameters

Cepstral coefficients can be obtained from the quantized log area ratio parameters (Q-LAR) that are developed in the course of GSM coding. To achieve this, the Q-LAR parameters are first transformed into the corresponding LPC coefficients, from which cepstral coefficients are generated using the approach described in Expressions 3.11 and 3.12 [2]. As was mentioned in previous chapters, since the LPC is of order 8, the LPC-derived cepstral coefficients (from Equation 3.11) contain no new information for cepstral indices greater than 8. In Figure 4.2 this process involves the process boxes labeled “LPC analysis” and “LPC cepstra front end”.

The quantization that takes affects the LPC filter information is performed in the log area ratio domain. As we have mentioned before, the reason for having quantization in the LAR domain is that this representation presents better robustness properties for quantization. From Equations 3.8 and 3.9, we can see that if the quantization is considered to be an additive noise term in the LAR domain, the effect on the LPC coefficients will be cumulative in the LPC domain: this means that the quantization noise will accumulate towards the higher order coefficients as a result of the recursions of Equation 3.8. Because the relationship between the LPC and cepstral coefficients is also a recursion, the quantization noise will accumulate in the higher order cepstral coefficients.

In order to illustrate this property of the quantization noise introduced to the cepstral coefficients due to LAR quantizing and coding, we use a normalized mean squared error metric (NMSE). For the j^{th} cepstral coefficient, we define to be:

$$NMSE[j] = \sum_{i \in Corpus} \left(\frac{(\mathbf{s}_i[j] - \hat{\mathbf{s}}_i[j])}{\mathbf{s}_i[j]} \right)^2 \qquad (4.9)$$

where $\mathbf{s}_i[j]$ is the j^{th} cepstral coefficient of the cepstral vector derived from the unquantized LAR parameters at the speech frame i and $\hat{\mathbf{s}}_i[j]$ is the corresponding coefficient derived from the quantized LAR parameters. Figure 4.3, shows on the left panel the NMSE for the 13-order cepstral coefficients computed across the TIMIT corpus comparing cepstral pairs derived from unquantized and quantized LAR coefficients. As expected, we can see that the normalized distortion follows an increasing trend as a function of the cepstral coefficients. In later parts of this chapter we will make use of these patterns to circumvent the effect of quantization on recognition.

4.2.2 Deriving cepstra from the residual signal

Cepstral features can also be generated from the short-term residual signal. In a process similar to deriving cepstra from LPC information, residual cepstra can be derived before and after the residual signal has been quantized and coded (*i.e.*, cepstra derived from short-term residual or from reconstructed short-term residual). In both cases, the signals involved are time-domain signals. Thus cepstra can be derived either in the conventional MFCC way or using the LPC-cepstra method. The corresponding path that this process follows in Figure 4.2, goes through the blocks labelled “LPC analysis” to “RPE-LTP codec” and finally to “RPE-LTP cepstra front-end”. Unlike the quantization process of the log area ratios, the quantization of the RPE and the LTP coefficients distorts the resulting reconstructed signal in a way which is difficult to characterize in any either the time, spectral or log-spectral domain.

Because we are working with the time-domain residual signal it is possible to perform an LPC analysis of order higher than 8. An LPC analysis of order higher than 8 would be desirable given that the short-term residual information has been removed from the residual signal through the LPC analysis. The remaining long-term information in this signal is contained in the long-term correlations and would be reflected in LPC coefficients of order higher than 8. We computed the corresponding NMSE between pairs of LPC cepstra derived from the short-term residual and the recon-

structed short-term residual in the same way we did for the LAR coefficients. In this case, we employed an LPC analysis of order 13. The results of the NMSE comparison are plotted in Figure 4.3 in the panel on the right. We can see that the pattern of distortion introduced by RPE-LTP quantization and coding affects mostly the lower cepstral coefficients. This pattern is the opposite as that observed from LAR quantization.

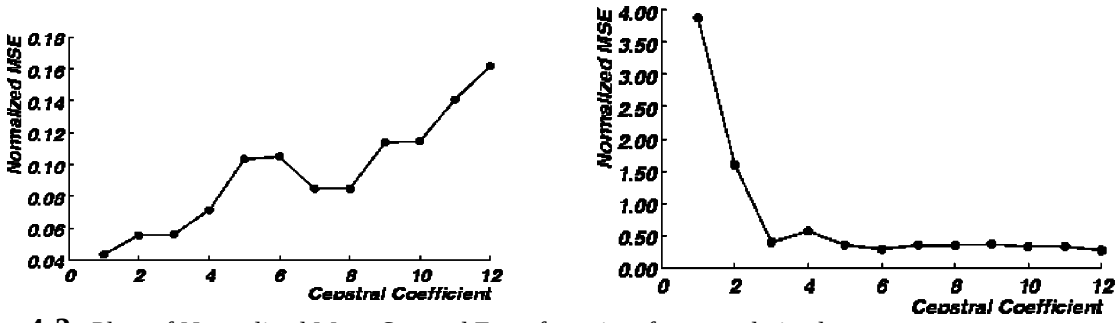

Figure 4.3 Plots of Normalized Mean Squared Error for pairs of cepstra derived from quantized/coded and unquantized/uncoded GSM parameters. The panel on the left depicts the comparison of LPC cepstral data and the panel on the right the short-term residual cepstral information.

4.2.3 Deriving cepstra from both residual and LPC information

Due to the limited order of the LPC analysis of the full rate GSM codec it is desirable to include the information contained in the LAR and in the RPE-LTP coefficients in the feature vector used for recognition. There are two ways to achieve this. The first method consists of reconstructing the short-term residual signal and then filtering this signal through the LPC synthesis filter, followed by the generation of the cepstra from it. The second way is to derive separately the cepstral feature associated with the LAR coefficients and the cepstral features associated with the reconstructed short-term residual and adding the two cepstral streams. Because of the homomorphic property of the complex cepstrum [65] the resulting feature is equivalent to that of the convolution of the impulse response of the filter with the reconstructed short-term residual time series.

In order to achieve the maximum possible recognition accuracy we need to include information contained in both the short-term and the long-term parameter streams in

the GSM codec. Because the patterns of distortion due to coding in the corresponding cepstral features derived from these streams are different, it is reasonable to expect that instead of simply adding the two cepstral streams it would be more advantageous to perform a weighted sum of the two sources of information. We will discuss this idea in section 4.4.

4.3 Effects of GSM coding on speech recognition accuracy

In this section we describe the results of a series of speech recognition experiments using cepstral features derived from the reconstructed waveforms and from the GSM parameters before and after quantization and coding. By doing this we will explore the answers to the following two questions:

- How much information relevant to ASR there is in each of the two main GSM parameter streams (LAR and RPE-LTP)?

- How much is GSM coding affecting each of those two parameter streams from the recognition point of view?

The recognition experiments in the following subsection were performed using the reduced bandwidth and downsampled version of the speaker independent component of the Resource Management RM1 corpus [9]. In order to explore the behavior of these cepstral streams in word accuracy terms under additive noise we also performed experiments adding white Gaussian noise at approximately 18 dB SNR. The acoustic models employed for recognition consisted of a set of 2500 senonically tied continuous density HMMs modeled by two Gaussian densities per mixture. The language model was based on bigrams trained on the training data set.

4.3.1 Contribution of GSM feature streams to recognition and the effect of quantization on ASR accuracy

Table 4.1 below shows recognition word error rates obtained using the two types of cepstral feature sets described in the previous subsection, with and without quantization and coding, as well as with and without additive noise present. For each feature set, acoustic models were trained with matching features, but with no additive noise present. The first three rows indicate the baseline experiments done using the standard cepstral analysis of the uncoded speech (first row), GSM-coded test only (*i.e.*, mismatched conditions, second row) and GSM-coded train and test conditions (*i.e.*, matched conditions, third row). We can see that the effect of GSM coding on recognition error rate was relatively moderate for this data set: the error rate displays a relative increase of 20% for clean speech and of 6% for noisy speech with mismatched training, and increasing about 5% under matched training conditions.

Rows 4 through 7 of the same table compare recognition accuracies obtained using cepstra generated from unquantized and quantized LARs. The table also compares results obtained with cepstra derived from the original short-term residual signal with those obtained using cepstra derived from the reconstructed short-term residual signal. The word accuracy obtained with these pair of features indicates that the residual signal contains information useful to recognition. This information is missing from the LAR parameters limiting its recognition performance: we can see from this table that the accuracy obtained using cepstra derived from LAR information is approximately 20% worse than the baseline recognition accuracy obtained with MFCC features.

We can also see from the same table that the increase in word error rate due to quantization of the LAR is smaller than that due to RPE-LTP coding. There is an absolute difference of 0.4% between the LAR and Q-LAR results, amounting to approximately 4% relative increase in the word error rate. Between the short-term residual and the RPE-LTP conditions there is a 3.6% absolute difference which amounts to more than 10% relative increase.

We can conclude from the analysis presented in this section that there is still information that is needed for optimal ASR left in the short-term residual signal. At the same time, quantization and coding in the short-term residual domain introduces a lager degradation in accuracy terms. This happens in spite of the fact that the percentage of bits allocated to represent the short-term residual is much larger than the bits representing the LPC information. In the next subsection we describe recognition experiments aimed at mixing back the information of the LAR coefficient with the short-term residual information in a way that tries to minimize the effect of GSM coding and quantization in the error rate.

Feature Set	Clean	AWGN
MFCC	**10.3%**	55.0%
MFCC (mismatched models)	**12.3%**	58.5%
MFCC (matched models)	**10.8%**	52.5%
LAR Cepstra	12.1%	45.9%
Q-LAR Cepstra	12.5%	55.1%
Short term residual Cepstra	28.9%	98.6%
RPE-LTP Cepstra	32.5%	96.1%

Table 4.1 Word error rate results for Resource Management recognition experiments using standard cepstral features (rows 1, 2 and 3) and features derived from GSM parameter streams without and with quantization under clean acoustic conditions and additive white Gaussian noise conditions.

4.4 Recombining LPC and RPE-LTP information

As we mentioned in the subsection above, according to the traditional LPC approach, reconstructed speech waveforms are obtained by the convolution of the impulse response of the LPC synthesis filter with the reconstructed short-term residual signal. The cepstrum of the speech waveform can be estimated by adding the cepstra of the LPC filter to those of the residual [65]. As discussed previously, however, the NMSE of these two sets of cepstral coefficients present different trends as functions of

the cepstral coefficient, suggesting that there might be better solutions than a plain vector addition of these features. In this section we show that we can improve recognition accuracy by selectively combining Q-LAR-derived cepstral coefficients with cepstral coefficients derived from the GSM reconstructed residual signal.

We consider two ways of combining the cepstra representing the LPC filter and the residual filter: (1) direct addition of the two sets of cepstra (which corresponds to convolving their time series) and (2) assembling a 13-dimensional composite cepstral vector by concatenating a subset of the cepstral coefficients representing the LPC filter with a subset of the cepstral coefficients representing the residual waveform. In this approach the goal is to incorporate the overall information in the recognition feature, while trying to avoid the regions of large distortion. Considering a weighted sum of the cepstral vectors as a general solution to the problem reduces it to finding the best weighting coefficients. Assembling a recognition feature from subsegments coming from the LAR cepstra and from the short-term residual cepstra can be interpreted to be a special case of a weighted sum, in which the weighting coefficients are either ones or zeros. We implemented this procedure by combining the first j coefficients of the quantized LAR cepstra and the last $13-j$ coefficients of the reconstructed short-term residual cepstra. Thus the problem is reduced to finding a cutoff parameter j.

The table below compares recognition results for a set of values of the cutoff parameter j, ranging from $j = 5$ to $j = 10$, plus the cases where $j = 0$ and $j = 13$. We note that in this table a cutoff of zero is equivalent to using a 13-element GSM reconstructed short-term residual MFCC vector, while a cutoff of 13 is equivalent to using LPC derived cepstra. From this table, we can observe that the best results are obtained when approximately 8 cepstral coefficients representing the LPC filter (including the

energy coefficient c0), are concatenated with 5 coefficients representing the residual signal.

Cutoff	Clean
0	32.5%
5	11.2%
6	10.8%
7	10.3%
8	10.3%
9	10.4%
10	11.3%
13	12.5%

Table 4.2 Recognition results for Resource Management experiments using different values of concatenation cutoffs.

The table below compares recognition error rates obtained in the following scenarios: adding the cepstral vectors, finding the cepstral feature of the convolved signal (or restored speech), and the best results obtained through concatenation (*i.e.*, best results from table 4.2). As can be seen, the concatenated feature vector is more effective than simple addition or convolution. More interesting is the fact that the recognition accuracy obtained using the concatenated GSM feature vector is greater than the accuracy obtained using coded or reconstructed waveforms and equal to that of the original uncoded speech waveform.

Feature	Clean
MFCC (no GSM)	10.3%
GSM (matched models)	10.8%
Convolution	10.8%
Sum	10.9%
Concatenation (cutoff = 8)	**10.3%**

Table 4.3 Summary of Word error rates for Resource Management for baseline conditions and three methods of combining short-term residual and LPC information into recognition feature.

4.5 Summary

In this chapter we have extended the analysis of the full rate GSM codec that we started in Chapter 3. We saw that the RPE block is the source of the distortion introduced to the reconstructed short-term residual. This distortion is a function of the long-term residual, and can be thought of as a non stationary additive noise. Because of this non-stationarity we cannot gainfully apply compensation methods which are based on stationarity assumption of the additive noise (such as CDCN or spectral subtraction).

We also saw in this chapter that the effect of GSM distortion can be analyzed from the perspective of its effect on the Gaussian densities, as well as from the perspective of its effect on speech recognition. We performed experiments based on quantized and unquantized versions of the codec parameters. In spite of having 85% of the bit-rate in the GSM codec allocated to represent the short-term residual information, we saw that the recognizer suffers the largest degradation from the quantization introduced to the residual signal. The parameters representing the short-term residual seem to carry information that is less important for recognition than the LPC information. This case is particularly true with the full rate GSM codec where the LPC analysis is of order 8. Based on this observation we showed that we can avoid the impact of GSM coding on recognition if we avoid reconstructing the overall speech signal and if instead we generate a recognition feature directly from the GSM codec parameters. We proposed doing this by selectively weighting the contribution from each of these two domains to the overall feature. In this way we can avoid the effect of quantization and coding on recognition to a large extent. It is not necessarily always the case that the recognizer has access to the GSM codec parameters. More commonly, the speech recognizer will have access only to the reconstructed speech signal. In these situations it is impossible to separate the distortion introduced by the RPE-LTP block (which affects the lower cepstral coefficients) from the distortion introduced by the LAR quantization block (which affects the upper cepstral coefficients). In the remainder of this dissertation we will explore approaches to situations where only the reconstructed speech signal is available.

Chapter 5
Phonetic-class based RPE-LTP distortion modeling

In previous chapters we have described the basic operation of the GSM codec and its effect on speech recognition. We have also explained the basic idea behind long-term predictive coding, and its effect on speech recognition. In Chapter 4 we proposed employing the speech codec parameters directly in order to circumvent the effect of quantization and coding on ASR performance. Evidently this would imply that the recognizer has access to the codec parameters. In the topology where an ASR application is connected to a mobile user through a wireless network and a PSTN connection, it is unavoidable that the speech signal is reconstructed before it reaches the ASR application. Therefore it is necessary to focus on the speech signal, as opposed to the codec parameters.

Based on the analysis of the operation of the GSM codec, we observed that in the process of coding the short-term residual the RPE block introduces a significant level of quantization distortion that affects ASR accuracy. The energy in the quantization error $\Upsilon(r[n])$ is a function of the energy of the long-term residual, which in turn depends on how well the LTP block can predict the current subframe. Based on our knowledge of the speech production mechanism, we know that some phonetic categories are likely to be predicted more accurately by the LTP process: the phones whose signal normally show high periodicity. Intuitively, one can expect that the RPE-LTP coding process of the short-term residual will introduce different patterns of distortion to phones with different characteristics. In this chapter we analyze this idea.

5.1 RPE-LTP induced spectral distortion

In order to establish a relation between phonetic identity and amount of distortion introduced by coding (specifically, by the RPE coding block) we must specify a metric that will reflect the degradation between the long-term residual and its quantized approximation. In this section we present such a metric and describe the distributions we obtain when applying it to the TIMIT database,

We use the relative log spectral distortion (RLSD) distance to measure the dissimilarity between the reconstructed and the original innovation (or long-term residual) sequences for each frequency ω. If $S(\omega)$ represents the power spectrum of a long-term residual subframe and if $S_R(\omega)$ represents the power spectrum of the corresponding quantized long-term residual subframe then, the RLSD between both signals is:

$$RLSD = \frac{1}{\pi}\int_0^{\pi}\left|\frac{\log(S(\omega)) - \log(S_R(\omega))}{\log(S(\omega))}\right| d\omega \qquad (5.1)$$

As can be observed, this metric reflects the ratio of the differences between the distortion introduced to the log-power spectra of the long-term residual by the RPE block, normalized for each frequency by the magnitude of the log power spectra of the original signal. When no distortion is introduced, both power spectra are equal and the RLSD is zero. The RLSD can be thought of as a type of frequency-averaged inverse SNR.

We computed the relative log spectral distribution produced by the RPE-LTP codec on a subset of the training utterances of the reduced-bandwidth, downsampled TIMIT corpus. In order to observe the difference between the incoming and outgoing signals of the RPE block, we modified the GSM codec to produce output files containing the samples associated with these two signal streams for each subframe. We then computed

the log power spectra of the two streams' subframes and then computed the relative log spectral distortion as described above.

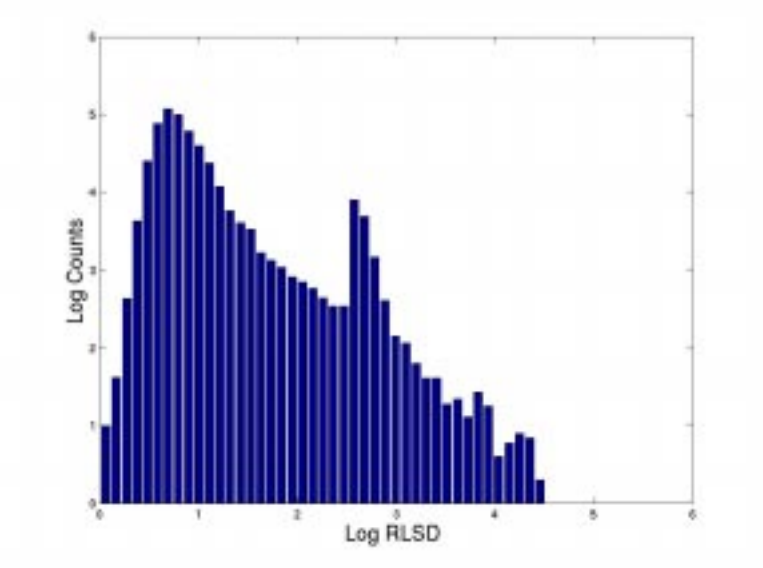

Figure 5.1 Log-histogram of the log-RLSD observed in a potion of the training part of the TIMIT corpus.

Figure 5.1 shows a histogram of the logarithm of the frequency of the values of the RLSD of this corpus. The horizontal axis represents the amount of log RLSD observed per codec subframe (*i.e.*, 40 samples).

The log-counts are roughly clustered around two regions separated by approximately the value of 2.8. It should be noted that many of the frames with the greatest RLSD are regions of silence, or near silence, within speech for which $S(\omega)$ can be relatively small in magnitude compared to the amount of distortion introduced. The majority of the frames suffer only a relatively moderate amount of distortion, so most of the instances the LTP section of the codec is able to produce a short-term residual estimate that resembles the actual short-term residual in a reasonable way, thus making the resulting long-term residual's energy small.

In order to establish and analyze the relation that exists between the degradation in recognition accuracy and the amount of RLSD introduced by the RPE-LTP block, we

performed two phonetic recognition experiments: (1) testing using clean speech data and (2) testing using speech that had undergone GSM coding. In both cases the acoustic models were trained using clean speech. We computed the phonetic error rate of both experiments, both for when GSM coding is present and when it is not. From this information we were able to compute the percentage of increase in error rate for each phone due to GSM coding. We also computed the average value of the RLSD associated with each of the recognizer's feature frames (*i.e.*, 10 ms. of speech) of every phone. This was performed based on the phone segmentations of the TIMIT corpus (TIMIT is a phonetically labeled corpus) and the modified version of the GSM codec we described above. In Figure 5.2 we present a scatter plot in which we combine both recognition and RLSD information for every phone: the horizontal axis represents the average of the log of the RLSD per phone, while the vertical axis describes the relative increase in WER due to GSM coding. We can see that the phones that incur an average log RLSD value of about 2.6 or below have a degradation in error rate of 20 percent or less. These phones are mostly vowels (**ae**, **eh**, **ah**, **aw**, **aa**, **ay, uh** etc). There are phones with log RLSD values between 2.6 and 2.8 whose relative degradation go above 20 percent. Finally, we can see that the consonants **f**, **z**, **v**, and **dh** have log RLSD above 2.8 and suffer a degradation of over 40 percent in error rate. Informally, from Figure 5.2 we can observe a loose relation or trend between the average RLSD observed in a given phone and the increase in its corresponding phonetic recognition error rate due to GSM coding, as well as between phonetic classes and observed average RLSD.

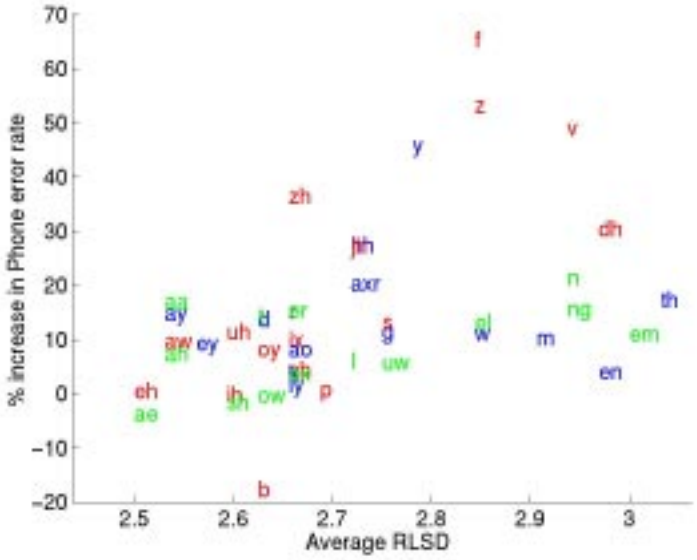

Figure 5.2 Scatter plot of the phonetic units of the TIMIT corpus, according to their average RLSD and their relative increase in phonetic recognition error rate due to GSM coding.

5.2 Relating relative log spectral distortion of the long-term residual pattern to phonetic classes

In the previous subsection we described the RLSD metric and showed that there seems to exist a relation between the mean RLSD observed at the subframes corresponding to a phone and the phonetic identity and increase in WER. In this subsection we extend this analysis of phonetic accuracy based on histograms of the RLSD of the long-term residual.

We constructed histograms of the log counts of the logarithm of the values of the relative log spectral distortion for each of the 61 phonetic units of the TIMIT database. Having obtained such histograms we normalized their areas in order to account for the differences in frequency of occurrence of the phonetic units. We grouped these units into phonetic clusters by incrementally clustering the two closest normalized histograms, (where the distance between to histograms was defined as the sum of the

squared difference between the values of each bin). The clustering process was started by placing each phonetic element in a class of its own, and finding the closest two histograms. After finding them, the normalized histograms are added and renormalized, which is equivalent to computing the geometric mean of the bins of both histograms as we are representing the logarithms of the counts. The Table 5.1 below shows the cluster obtained when the process was stopped at 8 classes. It also shows the average phonetic error rate per class with and without GSM coding and the corresponding relative degradation in error rate. In describing the clustering we refer to the categorization and description of the phones included in the TIMIT document (*cfr.* [61])

	Class Members	**Class Phone Error rate**	**Class GSM Phone Error Rate**	**% Degradation**
1	hh jh dx b d g k ch p t	32.4%	36.7%	13.27%
2	ng m n em en v eng	29.3%	42.1%	43.7%
3	h# bcl dcl gcl kcl pcl tcl pau	N.A.	N.A.	N.A.
4	hv aa ae ah eh aw ay ey	41.9%	44.7%	6.68%
5	epi	N.A.	N.A.	N.A.
6	ix iy ow oy ux zh nx ao ih r er ax sh uh	35.9%	38.5%	7.24%
7	dh th q	47.3%	64.3%	35.94%
8	axr uw f l ax-h el s w y z	30.9%	39.5%	27.83%

Table 5.1 Phonetic classes generated by automatically clustering phone distortion histograms and their corresponding phone recognition error rates without and with GSM coding.

Class 5 in Table 5.1 corresponds to the segments labeled as epenthetic silence, described in the TIMIT documentation as generally found between a fricative and a semivowel or nasal. Because this symbol does not appear in the phonetic dictionary used for recognition, no phone error rate is associated with it. Similarly, Class 3 grouped the closures for all the stops **b, d, g, k, p**, and **t**, as well as the utterance begin and end markers and the pauses, **h#** and **pau**. Because pronunciations in the dictionary don not explicitly have the closures indicated, no phone error rate is associated with them either.

Class 1 includes all the stops except **q**, both the affricates **jh** and **ch**, and the semivowel **hh**. Class 2 encompass all the nasals except the nasal flap **nx**, but includes the fricative **v**. Classes 4 and 6 split the vowels; Class 4 includes also the voiced **h**: **hv**, and Class 6 the fricatives **sh** and **zh**, the nasal **nx**, and the glide **r**. Class 7 includes the fricatives **dh** and **th**, as well as the stop **q**. Class 8 is the most heterogeneous, and includes fricatives, semivowels, and vowels.

Class 7 has the highest absolute class phone error rate without GSM coding, and Classes 1,2 and 8 have the lowest. Classes 2 and 7 are the classes that suffer the greatest amount of relative degradation when GSM coding is introduced, and Classes 4 and 6 are the most robust to GSM coding. We can see that the use the distribution of the RLSD introduced by the RPE-LTP in the form of normalized histograms for the purpose of clustering the phones produces classes with some phonetic homogeneity. We also note that classes dominated by vowels suffer the least from GSM coding, while groups dominated by nasals, fricatives, and some other consonants suffer substantially larger relative degradation in their class phonetic error rates due to GSM coding.

5.3 Summary

In this chapter we have defined the RLSD metric and observed its distribution for the overall speech signal, and for the different phonetic families. This RLSD is computed based on the differences of the log-power spectra of the long-term residual and the reconstructed long-term residual. We saw how based on distribution patterns of the RLSD, one can establish reasonable phonetic categories. In other words, the GSM codec presents consistent distortion patterns according to the phonetic properties of the signal it encounters. This dependency on phonetic identity can be exploited to achieve robustness to GSM coding. We will deal with this topic in future chapters.

Chapter 6
Weighted acoustic modeling for robust ASR in GSM codec environments

The RPE section of the GSM codec introduces distortion (the RLSD) that is associated with the phonetic characteristics of the speech signal such as its periodicity and LT predictability, and thus presents similar patterns or distributions for phones under similar phonetic groups. Some phones receive on average higher amounts of RPE-induced RSLD. As a result of the dependence of distortion on the phonetic characteristics of the signal, some phonetic categories are more robust to RPE-induced distortion and their recognition is affected less by GSM coding. We have seen that the features used in speech recognition as opposed to speech coding, do not use long-term information or long-term correlations of the speech signal. Even though the long-term correlation and long-term predictability features contain little information of utility for recognition, we have seen in Chapter 5 that this predictability determines the amount of distortion introduced by the codec, and thus we need to include this information during the recognition process of coded speech.

In this chapter we attempt to capitalize on these observations in order to achieve better recognition accuracy under GSM coding. The approach we propose in this chapter is that of acoustic modeling based on distortion classes.

6.1 Weighted acoustic modeling for HMMs

In Chapter 5 we observed that not all the phones in an utterance undergo the same amount of relative log spectral distortion. Less degradation in recognition accuracy is observed in general for phones that undergo less distortion. It is expected that the phones, and in general the speech regions, that undergo little or no distortion should be

modeled by HMMs that have been trained using data that received little or no distortion.

Similarly, for those phones for which GSM coding produces a larger average distortion, we expect to obtain greater recognition accuracy if during recognition we employ models that are trained with data that has undergone a greater amount of distortion. Intuitively, we can think of the distortion introduced by the GSM codec as a non-stationary additive noise whose intensity (*i.e.*, energy) is inversely related to the long-term predictability of the speech signal. Then, we would like to perform speech recognition based on acoustic models that match the instantaneous distortion conditions of the speech observed.

One possible way to implement this type of acoustic modeling is to use two acoustic models during recognition: one derived from clean speech and the other from noisy or distorted (*i.e.*, GSM) speech. This idea can be extended to the scenario of having several acoustic models in which each model has been trained under different levels or intensities of noise or distortion. Figure 6.1 illustrates this concept.

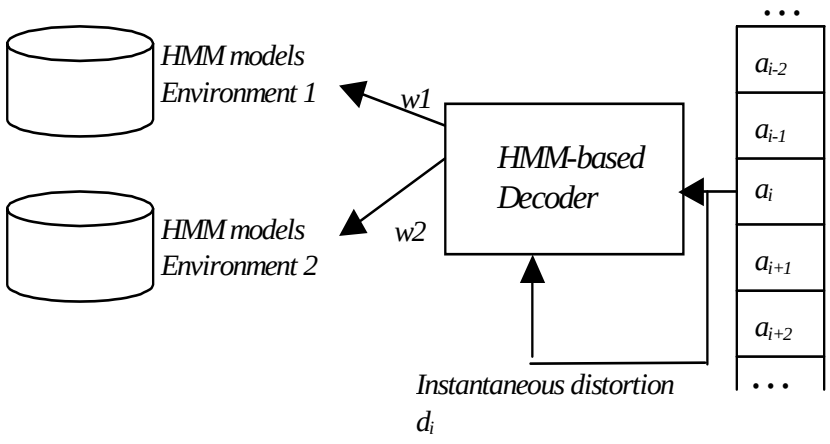

Figure 6.1 Multiclass weighted acoustic modeling based on instantaneous distortion estimates.

As previously mentioned, our aim is to combine two sets of models: one derived from undistorted data and one from distorted data. In the remaining sections of the

chapter we will further describe the possible ways to implement this proposed technique, the strategies to obtain the weighting coefficients as well as how the models representing the different distortion environments can be trained.

6.2 Combining acoustic models by means of mixture weighting

In this section we introduce the two main methods by which acoustic models can be combined. The first method works at the log likelihood level and the second works at the Gaussian mixture level (or pdf level). We provide more details of these two methods:

The first method is to consider several models when recognizing an utterance is by combining the posteriori probabilities obtained from these models into a log-linear posterior probability distribution. Equation 6.1 describes the probability of a string W given the observation features A, expressed in terms of the posterior log probabilities of the N different models. These scores are weighted by the terms λ_j and incorporated into a log-linear score expression:

$$p(W|A) = \exp\left\{\log(C(\Lambda)) + \sum_{j=1}^{N} \lambda_i \log p_j(W|A)\right\} \tag{6.1}$$

Where $\log p_j(W|A)$ represents the log-likelihood that is computed using the j^{th} model, the sequence of observation features A and the hypothesis word string W. The term $\log(C(\Lambda))$ is a normalization factor.

A different approach, and the approach followed in this thesis, is to achieve this model combination is by merging the weighted distributions of both models into a sin-

gle set of models. This is the method we follow because it can be implemented easily by means of simple modifications of the training and recognition code.

As we explained in Chapter 2, the modeling of the probability density functions associated with the HMM states is typically done in terms of Gaussian mixtures. So the probability that state q_i emits the i^{th} observation vector a_i is expressed as

$$p(a_i|q_i) = \sum_{k=1}^{K} p_{q_i}(k) N(a_i, \mu_{q_i,k}, C_{q_i,k}) \tag{6.2}$$

The term $p_{q_i}(k)$ is the prior probability of the k^{th} Gaussian component of the HMM state q_i. For a given state, the sum of these terms $p_{q_i}(k)$ over all k is equal to 1. The terms $\mu_{q_i,k}$ and $C_{q_i,k}$ represent the mean vector and the Covariance matrix, respectively.

We consider the amount of distortion that a frame or a phonetic class undergoes by evaluating the posterior probability using M different models associated with these distortion regions. We express this concept by introducing a function f that weights the k^{th} posterior probability of the m^{th} model depending on d_i, the distortion of the observed frame a_i, and the identity of the state q_i. In weighted acoustic modeling, each state q_i now consists of the weighted Gaussian mixtures belonging to each of the M different corresponding environments $q_{i,m}$. The function f depends on the prior probabilities of the model $p_{q_{i,m}}(k)$. This function can be considered to be a mapping from the original set of prior weights $p_{q_i}(k)$ to their weighted versions according to amount of distortion observed. The resulting expression for the posterior probability becomes

$$p(a_i|q_i) = \sum_{m=1}^{M} \left(\sum_{k=1}^{K} f(p_{q_{i,m}}(k), d_i, q_{i,m}) N(a_i, \mu_{q_{i,m},k}, C_{q_{i,m},k}) \right) \quad (6.3)$$

In the expression above the new weighting factors $f(p_{q_{i,m}}(k), d_i, q_{i,m})$ are functions of the observed instantaneous distortion and of the identity of the state.

In the following sections we describe two special cases for performing weighted acoustic modeling. In the first case, the function f is factored into the prior weights, and a weighting factor that is a function of *average distortion* observed by the base-phone represented by the HMM. In the second case, the function f is also factored in two terms: one related to the prior weights of the mixtures of each state, and a second term that weights each mixture component of each observed feature vector a_i depending on the distortion class these mixtures represent according to an estimate of the *instantaneous distortion* affecting the i^{th} acoustic observation.

6.3 Weight factors based on average distortion

When the function f is independent of the prior weights $p_{q_{i,m}}(k)$ (*i.e.*, it scales these probabilities by some multiplicative constant), and depends only on the average distortion associated with state q_i, it can be expressed as the product of two terms

$$f(p_{q_{i,m}}(k), d_i, q_{i,m}) = p_{q_{im}}(k) \lambda^{(m)}_{q_i} \quad (6.4)$$

The coefficients $\lambda^{(m)}_{q_i}$ are the weights that, for each HMM state q_i, rescale the mixture component of every one of the m corresponding distortion environments according to some statistics of the distortion observed by the HMM state (such as, the

average phonetic distortion observed by the basephone that the state is representing). For the particular case when only two distortion classes exist (*e.g.*, clean and GSM-coded speech), only one λ is necessary to define the weights of this model (as we can define $\lambda^{(2)}_{q_i} = 1 - \lambda^{(1)}_{q_i}$). After factoring out these rescaling terms Equation 6.3 can be rewritten for the two-class case as:

$$p(a_i|q_i) = \lambda_{q_i}\left(\sum_{k=1}^{K} p_{q_{i,1}}(k)N(a_i, \mu_{q_{i,1},k}, C_{q_{i,1},k})\right) + (1-\lambda_{q_i})\left(\sum_{k=1}^{K} p_{q_{i,2}}(k)N(a_i, \mu_{q_{i,2},k}, C_{q_{i,2},k})\right) \quad (6.5)$$

The new posterior probability expression can now be interpreted as a “mixture of mixtures” in which the component mixtures come from the different distortion environments and are weighted to constitute the overall state distribution based on some statistic derived from phonetic distortion information.

In practice, we implement the method described in this section in the following way: the set of HMM models representing the clean environment is trained using data recorded exclusively on non-GSM speech, and the set of HMM models representing the noisy environment is trained using GSM speech. The models are recombined off-line prior to recognition and no retraining process is involved.

Given that no retraining of the new models is involved, an important set of questions is raised here and will be explored in the next section:

- Why should recognition using models resulting from the combination of two sources, *i.e.*, models trained under “matched” conditions (GSM) and models trained under clean conditions, be any better than recognition using only matched condition models?

- Should not the Maximum Likelihood estimates of the HMM parameters and distributions (obtained through the Baum-Welch training algorithm) for the matched conditions result in the models that give the maximum overall likelihood of the training data? How can these maximum likelihood models be enhanced from data coming from mismatched conditions?

6.4 HMM sensitivity to phonetic perturbations

In the previous section we suggested the idea of combining two sets of acoustic models according to the average distortion observed by each phone. In this section we will try to answer the questions we raised at the end of the previous section in terms of the HMM training procedure, the Baum-Welch (BW) algorithm.

In Section 2.1 we reviewed the basic concepts behind HMM-based speech recognition. We mentioned that the evaluation of the probability of observing a set of feature vectors given a set of states was the sum of the probabilities of each possible trajectory of states (Equation 2.7). In other words, during the process of HMM model training and the process of decoding, all possible state trajectories associated with a particular HMM chain are weighted and considered in the overall score. This implies that speech data not only affect the models of the phones to which they belong but also those of adjacent phones and to a lesser extent phones which lie farther away.

Now let us consider a process in which the speech signal undergoes a certain phone-dependent distortion process which affects only a few phone categories in the corpus. If we retrain a set of HMM models that were developed from the original undistorted speech, because of the property we explained above of the BW algorithm, not only the models of the phones affected by the distortion process change but we also expect to see some effect on every single phone that appears in the utterances where distortion occurred.

In order to illustrate this phenomenon we performed a simulation experiment. We took the TIMIT corpus and trained context-independent HMM models with one Gaussian density per mixture based on clean speech. Then, using the segmentation provided with the database, we added a constant noise vector to the cepstral vectors of each of a given phone in the database. Afterwards, we ran a few iterations of the BW models but employing the modified (or perturbed) cepstra. At the end of the BW passes, we compared each of the means of the retrained acoustic models with the original acoustic models. We did this by computing the mean squared difference between the two mean vectors of both sets of models and adding this mean squared differences for each of the 5 states representing a phone HMM. In this way, we obtained the sum of the average distortion of the means of the 5 HMM states of each phone introduced by the perturbation of the data associated with a single phone in the database. As expected, perturbing a given phone affected the models of the other phones. Table 6.1 below shows a partial table (with not all the phones displayed) of results when the experiment was repeated for each of the phones in the TIMIT database. Values have been rounded to the closest integer. The columns indicate the phone to which the additive perturbation was introduced. The rows correspond to the phones whose HMMs were compared before and after retraining. For example, the column labeled **s** contains the values of the mean squared differences for the states of the HMM representing the phones on the rows when the phone **s** was perturbed. We can see that the largest value in this case is 47, that corresponds to the MSD between the states of the phone **s**. But also the models of the phones **ax** and **ix** get influenced substantially by this perturbation to this phone. Again, this is due to the fact that Baum-Welch considers every possible path along the sequence of states.

In the context of the GSM-coding-induced distortion problem, we saw in the last chapter that different phonetic distortion categories receive different amounts of average RLSD. The HMMs of the phones that receive little distortion are going to be affected, on average, in a proportion larger than the actual distortion observed by the phone. This, for example, can be seen in the example seen above where the phone **ax** received no distortion but was affected by the perturbation introduced to phone **s**. To

reduce this effect, then, an alternative is to use an interpolated model coming from the HMM models of distorted and undistorted data, in which each HMM model is weighted according to the amount of average distortion observed by the base-phone represented.

	aa	**ax**	**b**	**d**	**em**	**f**	**ix**	**l**	**s**
aa	49	0	0	0	0	0	0	0	0
aw	0	0	0	0	0	0	0	0	0
ax	1	31	2	1	0	23	30	23	26
b	2	2	197	2	2	2	3	2	3
ch	2	3	2	2	2	3	4	3	5
d	5	1	0	183	0	1	4	2	3
em	0	5	0	0	0	0	0	0	0
f	0	0	0	0	0	81	0	1	0
g	1	1	1	1	1	1	2	79	2
hh	4	2	0	0	0	3	6	2	9
ix	0	7	0	0	0	0	26	1	20
jh	3	1	1	1	1	1	5	1	2
l	0	0	0	0	0	0	0	21	0
s	1	1	0	1	1	1	0	1	47
v	1	3	1	1	1	1	2	1	1
zh	1	0	0	0	0	0	0	0	0

Table 6.1 Amount of distortion observed in target phones (rows) when a constant perturbation is introduced to a source phone (columns).

The forward-backward algorithm of the Baum-Welch technique is implemented using the beam search heuristic [55]. The beam search procedure consists of the evaluation of the overall probability that a sequence of states produces a sequence of observations is limited to the state paths (and thus phone boundaries) whose scores fall

within a certain multiplicative constant from the best path's score. The remaining paths get *pruned*. The multiplicative factor is inversely related to the beam width. Thus, this multiplicative factor is a parameter which can be used to control the amount of distortion that the Baum-Welch algorithm introduces to the HMM models.

In order to evaluate the effect of the beam width on the spread of phone-dependent perturbations affecting other phones, we repeated the experiment described above perturbing the instances of the phoneme **s** and retraining using 5 different values of beam-width. The results shown in Table 6.2 indicate the ratio of the sum of the distortion observed across all phones except phone **s**, divided by the distortion observed in the phone **s**, for different multiplicative factors used in the Baum Welch procedure. As we can see, reducing the values of the factor (by increasing the beam width) and thus considering lower scoring paths in the likelihood computation, tends to contribute to the spread of the perturbation introduced to the phone **s** into other phones.

Multiplicative factor	**Ratio**
1e-40	3.95
1e-60	4.73
1e-80	5.43
1e-100	5.89
1e-120	5.41

Table 6.2 Ratios of sums of distortion observed in phones other than s, divided by the distortion observed in phone **s**, when **s** is perturbed, for different beam widths used in training.

6.5 Weight factors based on instantaneous distortion

Equation 6.3 in Section 6.2 described the use of the function f for the purpose of weighting each of the mixture components according to the distortion associated to the observation. In Section 6.3 we discussed basing the weight on average phone RLSD (or some other statistic of the phonetic distortion), thus making this weighting independent of frame index i.

In general, however, the distortion introduced by the codec varies according to the long-term predictability of the time signal. Thus two different realizations of a phone can have very different instantaneous RLSD associated with them. We can extend the weighted acoustic modeling approach for the case where the weighting is based on the instantaneous distortion introduced by the codec. Then, if we separate the function f into two parts, one associated with the mixture weights of every distortion class (*i.e.*, $p_{q_{i,m}}(k)$) and the other associated with the probability of belonging to distortion class m given that the instantaneous distortion d_i was observed then

$$f(p_{q_{i,m}}(k), d_i, q_{i,m}) = p_{q_{i,m}}(k)\, p(m|d_i) \tag{6.6}$$

The weighting term $p(m|d_i)$ depends on the instantaneous distortion observed on the signal at frame i, which is associated with the observation a_i. This distortion d_i, is related to the quantization distortion $\Upsilon(r[n])$ introduced by the RPE (as discussed in Chapter 4), and thus related to the long-term predictability of the signal.

We can compute the weighting terms as a function of an estimated value of the instantaneous quantization distortion of the feature frame observed at frame i, $\hat{\Upsilon}_i$. The

weights then become $p(m|\hat{\Upsilon}_i)$. For the case of two distortion classes Equation 6.3 becomes

$$p(a_i|q_i) = p(m=1|\hat{\Upsilon}_i)\left(\sum_{k=1}^{K} p_{q_{i,1}}(k)N(a_i, \mu_{q_{i,1},k}, C_{q_{i,1},k})\right) + p(m=2|\hat{\Upsilon}_i)\left(\sum_{k=1}^{K} p_{q_{i,2}}(k)N(a_i, \mu_{q_{i,2},k}, C_{q_{i,2},k})\right) \quad (6.7)$$

where

$$p(m=2|\hat{\Upsilon}_i) = 1 - p(m=1|\hat{\Upsilon}_i) \quad (6.8)$$

In practice the codec is not providing any information about how good the RPE representation of the LTP signal is. We need to estimate this distortion and then map it to the weighting term $p(m|d_i)$. We will discuss in subsequent chapters how we can achieve this. Figure 6.2 shows a block diagram of a decoder based on the weighted acoustic modeling method described in this chapter, which in turn operates based on an estimate of the distortion term.

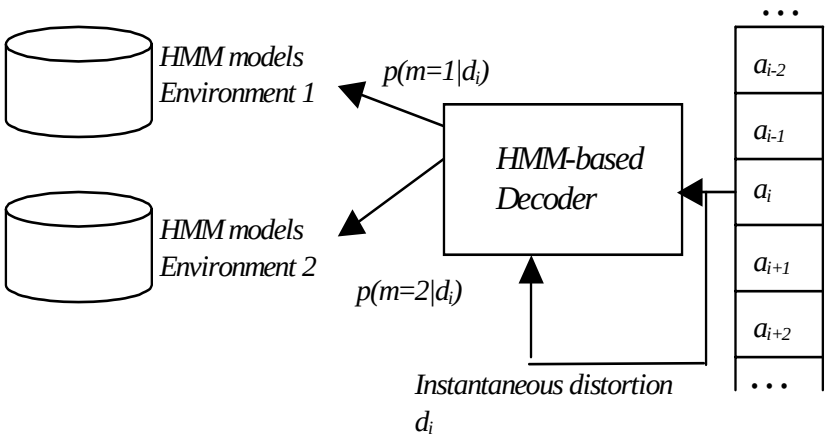

Figure 6.2 Block diagram representing a weighted acoustic modeling decoder which operates on the basis on an estimate of the distortion term.

6.6 Weighted acoustic modeling and other model combination techniques

Even though not proposed in the context of robust speech modeling under speech coding, it has been shown by other authors (*e.g.*, [57, 6]) that a recognition system's performance can be improved by combining several different acoustic models during recognition. We briefly describe the ideas behind these techniques based on composite acoustic modeling followed by a contrast of them with our proposed weighted acoustic modeling method:

- **Parallel model combination:** The aim of this technique [26] is to generate a set of parallel HMMs based on an estimate of the noise vector and a model of the degradation on the feature. The topology of the resulting HMM's is the same as that of the original HMMs. The additional models are generated from existing models and not retrained from additional distorted data.

- **Discriminative model combination:** Beyerlein focused on combining models (both acoustic and language models) at the log likelihood score level such that the discriminability of the classes is increased [6]. This technique has been successfully applied to the recombination of acoustic models in the Broadcast News domain and in multilingual recognition research efforts.

- **Multiple-stream parallel recognizers (such as subband-based recognition):** In this approach a set of parallel recognizers perform recognition on a set of independent acoustic features usually derived from different frequency bands and their individual hypotheses are combined into a single hypothesis. Depending on the level at which the recombination is performed, their HMM topologies can be more or less tied or can be anchored in time corresponding to segmental levels [7, 8]

As an extension to the multiple stream approach described above, Ming *et al.* [57] proposed the use of multi-model methods to integrate the capabilities of separate modeling techniques. They proposed the use of segmental and frame-based modeling, and define the state-dependent observation densities of the combined model as the product of the corresponding densities from each component model.

- **Deleted interpolation of acoustic models** The method of deleted interpolation has been employed in order to smooth the parameters of the distributions of discrete and semicontinuous HMMs in order to achieve a balance between detailed modeling and robust estimates of the parameters. Huang *et al* (*cfr.* [39])proposed a deleted interpolation method for continuous-density HMMs in which the likelihood obtained from context dependent and context independent models are interpolated. The interpolation weights in this method are estimated using a cross-validation technique as a maximum likelihood estimate would converge to the trivial and not useful value of 1.0.

In comparison, we have proposed a method for which the feature vectors have to be observed during training (as opposed to estimating its models, as in PMC) and an estimate of the distortion-class membership has to be estimated. Our method is based on a single feature contributing to different models, (as opposed to many features contributing to many models, as in multi-band recognition). The method we proposed uses a tied topology configuration in which the topology of the states of both models being combined is strictly the same. Instead, only the state's probability densities, which are modeled by Gaussian mixtures, are affected. This is in contrast to Ming's method that uses different structures of HMMs by "anchoring" them at some points. Finally, as opposed to Beyerlein's discriminative model combination (DMC), our method weights the state emission probabilities based on the information related to the distortion introduced by the codec process, which is a result of the effect of different degrees of long-term predictability present in the speech signal. DMC obtains the mixing weights by optimizing the discriminability of the classes, (*i.e.*, by minimizing the string error rate). Instead, our method derives the weights from distortion information.

6.7 Summary

In this section we proposed the technique of weighted HMM acoustic modeling to address the problem of codec induced distortion. This technique was postulated in two ways: when the combination weights are based on average statistics of the distortion observed by the phones and when the combination weights are based on instantaneous distortion estimates. In the next two chapters we will elaborate on these two approaches and evaluate their performance in terms of speech recognition accuracy.

Chapter 7
Weighted acoustic modeling based on average phonetic RLSD

In the previous chapter we introduced the idea of weighted acoustic modeling to achieve better recognition under GSM coding distortion. We mentioned that there are two ways in which the weighted modeling method can be implemented: making use of average distortion information and making use of instantaneous distortion information. We briefly introduced in that chapter some of the details of the algorithm based on average distortion statistics.

In this chapter we present the results of the recognition experiments when the weighted acoustic modeling technique is applied to recognition of GSM speech based on average distortion statistics. We will use measurements of the Relative Log Spectral distortion introduced to the long-term residual by the RPE coding block as our distortion statistics.

7.1 Effects of combining models trained separately on likelihood surfaces

The equation below represents the expression used for weighted acoustic modeling based on average phone statistics, for the case of two distortion classes

$$p(a_i|q_i) = \lambda_{q_{i,1}}\left(\sum_{k=1}^{K} p_{q_{i,1}}(k)N(a_i, \mu_{q_{i,1},k}, C_{q_{i,1},k})\right) \quad (7.1)$$
$$+ (1-\lambda_{q_i})\left(\sum_{k=1}^{K} p_{q_{i,2}}(k)N(a_i, \mu_{q_{i,2},k}, C_{q_{i,2},k})\right)$$

This equation can be rewritten in terms of the likelihood of observation a_i given the state q_i based on the clean model and the noisy model (clean and GSM, respectively):

$$p(a_i|q_i) = \lambda_{q_i} L_{Clean}(a_i|q_{i,1}) + (1 - \lambda_{q_i}) L_{GSM}(a_i|q_{i,2}) \tag{7.2}$$

Even though Equation 7.1 might represent to be a doubling of the mixture components that constitute the basic acoustic model (*i.e.*, 2*K* Gaussian components overall: with *K* Gaussians coming from the original clean models plus another *K* Gaussians coming from the GSM models). Since the models are separately trained, this does not create a consequent data insufficiency problem. Equation 7.2 suggests that we are interpolating or averaging both surfaces using the λ terms as weighting coefficients. We can think of models with larger number of Gaussian components as having a certain higher "resolution" or being less coarse models for the likelihood surface. By interpolating two surfaces of *K* Gaussian components, we are not gaining any more resolution. Instead we are generating a surface that might result in slightly smoother surfaces. This is particularly true when the means of the Model A that we are combining are close to with their counterparts in model B.

To illustrate this point, Figure 7.1 shows the contour plot and surface plot of three densities. The top row corresponds to a Gaussian distribution A, the middle row corresponds to a second Gaussian distribution B with means and variances similar (but not exactly equal) to those of distribution A. When we compute the surface of the distribution resulting from weighting models A and B (with weights 0.6 and 0.4, respectively) the resulting distribution produces a contour and surface plot displayed in the third row.

Strictly speaking, the resulting distribution is a bimodal weighted Gaussian mixture, however, because the means are so close to each other, the resulting distribution

looks like a single mode Gaussian with values in the variance matrix larger than those of distribution A or B. The resulting surface looks like distribution A or B “smeared”.

If we model the resulting surface with a parametric Gaussian distribution the estimated value of the mean will be between the value of mean A and mean B. The variances, or the diagonal entries of the covariance matrix, will be larger in value than their counterparts in the individual distributions A and B. If Gaussian A represents the clean data and B the noisy data, the mean of the mixture will be between the valued of the clean mean and the noisy mean.

In the case that our training produced over-fitted models *i.e.*, if we were training with insufficient data for the number of parameters we were estimating, and our models were modeling the set of training examples in a way that produced suboptimal decision boundaries, this normally would mean that the resulting variances are smaller or narrower than the variances of the actual distribution. Interpolating, or smoothing the surfaces, with its implicit broadening of densities (or increase in the value of variances) of the resulting model, might help to compensate for the effects of overfitting to a certain extent. This idea is also the motivation of techniques like deleted interpolation, as well as discount techniques in LM *n-gram* estimation [39, 45].

7.2 Tying the estimates of the weighting parameters

Equation 7.2 describes how a set of λs or weights should be associated with each state in the HMM model inventory. Usually a speech recognizer associates an HMM model with a phone with particular contexts (*i.e.*, a context-dependent phone or a triphone). In a medium-to-large vocabulary speech recognition task there are typically several thousand possible triphones. In order to ensure robust estimates and avoid data scarcity, states are tied based on phonetic decision trees [40], and sets of HMM states are shared or tied among these subsets of triphones.

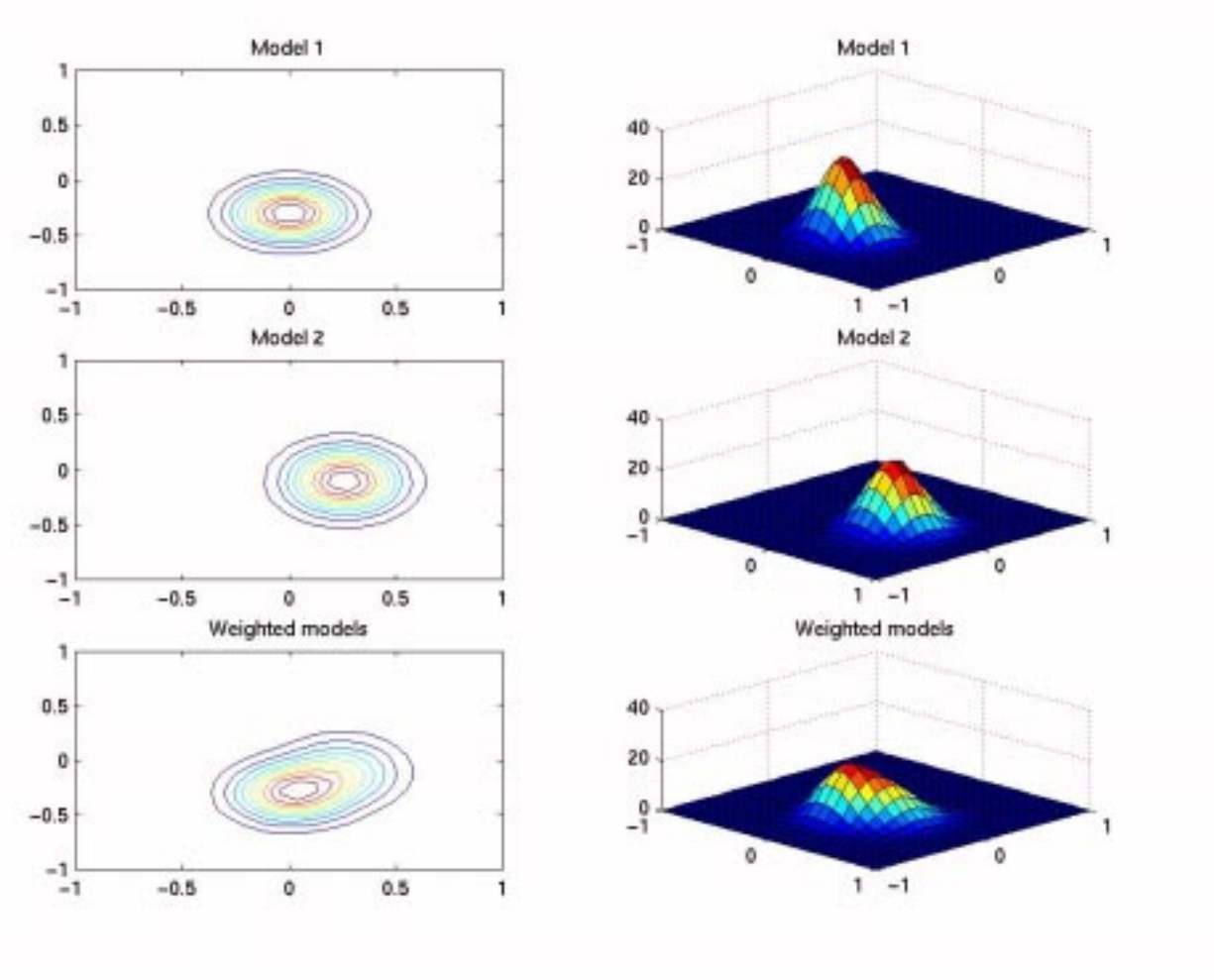

Figure 7.1 Interpolation of two likelihood surfaces through weighted acoustic modeling.

In the proposed weighted modeling approach based on average statistics, we might encounter the problem of having a large number of λs to estimate. To avoid this problem, we propose the use of tied λs: *i.e.*, sharing the λ parameters among different states based on phone category membership. We propose the following alternatives to achieve this: flat distribution of weights, phonetically-tied weights, and phonetic dis-

tortion-class based weights. We now present the details for each of these techniques, and leave the comparisons of recognition results to Section 7.4.

7.2.1 Flat distribution of weights

A single set of λ or weight parameters is assigned to the whole collection of HMM states. The new mixtures are going to be obtained using these "tied" distributions and thus the resulting likelihood surface will be the weighted average of both likelihood surfaces. For the case of two distortion classes: (*i.e.*, GSM and clean), there is only one free parameter $\lambda^{(1)}$, because $\lambda^{(2)}$ is equal to $1-\lambda^{(1)}$. As one of the λs, approaches the value 1.0, the other tends to approach zero and the recognition results will tend to match the results obtained by using the corresponding single set of HMMs.

Searching for the value that results in optimal recognition using a held out-set is now a simple task as the search is over a single parameter for the whole database.

7.2.2 Phonetically-tied weights

In this case the HMM states are grouped according to the base phone they represent and a single set of λs is assigned to each phone. For the case of two distortion classes, each phone has its set of λs defined as a free parameter. The number of free parameters is thus equal to the number of phones.

In order to find the set of λs, one can search exhaustively or estimate them from information contained in the histograms of the RLSD introduced to the phones. For the case of exhaustive search, it is generally better to reduce the number of free parameters by merging phones together into phonetic categories.

7.2.3 Phonetic distortion-class based weights

In Chapter 5 we observed that we could cluster the phones of a corpus based on patterns of distortion observed on each phone and obtain reasonable phonetic categories. We can use these automatically-derived phone clusters to tie the λ parameters in this way limiting the number of parameters to be estimated.

As in the case of phonetically-tied λs, the free parameters necessary to define the sets of λs can be either derived from the histogram of the phonetic class or searched exhaustively for the λs that maximize recognition rate on a held-out set.

7.3 Estimating the weighting factors from the relative log-spectral histograms

For a given set of phonetic clusters, we want to associate a set of weights such that each class will weight both models. These weights should be made proportional to the amount of distortion observed in each phonetic cluster. We propose to determine the values of these weights using the normalized log histograms of the relative log spectral distortion RLSD introduced in the long-term residual that we described in Section 5.1. To achieve this, we first cluster the phones into phonetic categories using the agglomerative method described in Section 5.2. We then obtain the normalized log-histogram of each resulting class and for each histogram we compute the value of the bin for which 50 percent of the counts have been accumulated, and then divide this value by the value of the highest bin. This bounds the value of λ between 0 and 1, which directly maps these into the desired range of values. In other words, we normalize it. Figure 7.2 depicts the 50% log counts of the log histogram of a certain class.

A value close to zero indicates that 50 percent of the counts are close to the low distortion area and that the associated weight λ will be small and $1-\lambda$ will be rela-

tively high (*i.e.*, close to one). In this case $1-\lambda$ should be the weight associated with the clean models, for this particular example.

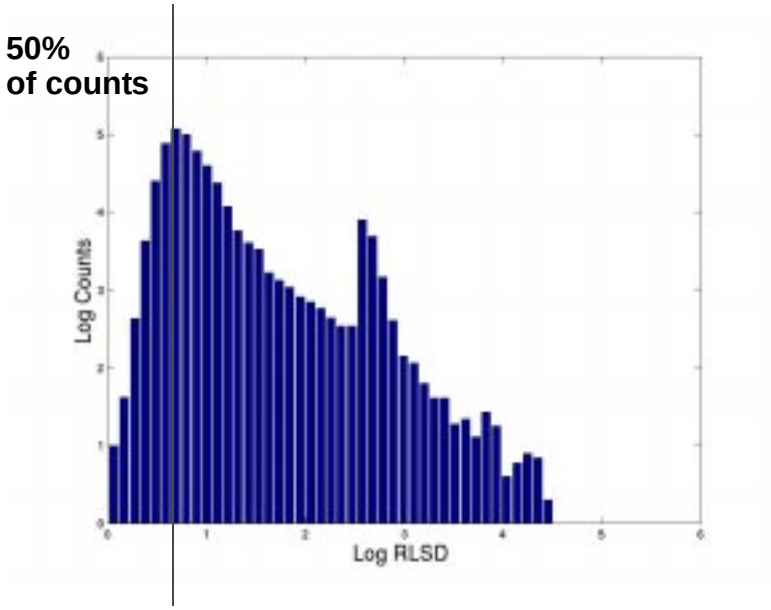

Figure 7.2 A histogram marking 50% of the log counts.

7.4 Recognition experiments

We performed recognition experiments on the TIMIT database using two distortion classes, one coming from clean data and one from GSM data. Table 7.1 shows the recognition accuracy *baseline* results obtained for the case of 8 Gaussians per mixture and the following training/testing conditions: Clean/Clean conditions, GSM conditions with mismatched (*i.e.*, clean) models, GSM conditions with matched models and GSM conditions in testing with training performed using both clean and GSM data. We observe that when both clean and GSM data are included in the training process the accuracy improves. This improvement can be associated with the use of more training data.

Test data	Training data	% WER
Clean	Clean	**11.5%**
GSM	Clean	13.0%
GSM	GSM	**12.2%**
GSM	GSM+Clean (multistyle)	11.9%

Table 7.1 Baseline recognition experiments for TIMIT database under different coding conditions.

Table 7.2 below shows the results obtained when weighted acoustic modeling techniques were applied. For all cases considered, two distortion classes were combined, one corresponding to clean data conditions and the other to the GSM conditions. We also considered different λ tying methods: flat distribution weights (equivalent to all λ pairs tied into a single class, as described in Section 7.2.1), 15 classes derived automatically employed the histogram-based procedure (described in Section 7.2.3), and 45 classes using phonetically tied weights (with a set of λs is associated to each base phone as described in Section 7.2.2). Table 7.2 refers to these weight-tying methods as 1 class, 15 class and 45 class, respectively in the column labeled “Number of phonetic categories”.

For each of these three tying scenarios there exist two methods of determining the value of the λ parameters. The first method, based on a held-out data set, searches exhaustively for the value of the λs that maximize the WER of this set. In order to facilitate the search, the best value for each λ can be computed independently of the others. The second method derives these weights by means of the histograms using the method described in Section 7.3.

For the exhaustive-search case, the values were initialized with the values derived from the histograms and the search was performed in the vicinity of these values, with one parameter being searched at a time.

The results in Table 7.2 show that the best recognition accuracy is obtained for the case of exhaustive search based on 15 phonetic classes. However for the case of histogram derived classes the results were just slightly worse. In the case of histogram-derived weights it did not make any perceptible difference whether there were 15 or 45 phonetic classes. These results are also slightly better than the 1-class (flat weights) exhaustive search of the parameters, which gives same results as multistyle training for this configuration of Gaussians.

The best word error rate in this case is 11.7, which is 0.5% better than the simple baseline case of matched GSM/GSM results. However, the relative reduction of the degradation introduced by GSM coding (*i.e.* the difference between 11.5% and 11.2%) gets is observed to be around 70% from 0.7 to 0.5%. This, again, was obtained by weighted modeling based on exhaustive search of weighting values of 15 classes. The classes were those derived automatically from the histograms.

Test data	Num. of dist. Cat.	Num. of phonetic categories	Weight search Method	% WER
GSM	2	1 class (flat w.)	Exhaustive search	11.9%
GSM	2	15 classes	Exhaustive search	**11.7%**
GSM	2	15 classes	Histogram derived	11.8%
GSM	2	45 classes (1 phone per class)	Histogram derived	11.8%

Table 7.2 Recognition experiments using various weighted acoustic modeling schemes and GSM coded speech.

The experiments described in Table 7.2 were based on the weighted combination of two source acoustic models. For those experiments, the total number of Gaussians used in recognition had been effectively doubled, making the decoding process computationally more expensive and thus slower. Nevertheless the extent to which improvement is possible regardless of computational complexity is an important consideration.

Another issue of importance is to determine the effect of doubling the number of parameters used to model the likelihood surfaces on the recognizer accuracy.

Figure 7.3 shows results of recognition experiments that employ the same phonetic clustering and λs obtained using 15 phonetic classes and exhaustively-searched weights as a function of the number of Gaussians per mixture. The horizontal axis is labelled according to the number of Gaussians per state in the source models used in the Clean/Clean and GSM/GSM cases in the top row (*e.g.*, 8 Gaussians), and according to the number of Gaussians in the resulting weighted acoustic models in the bottom row (*e.g.*, 8x2 Gaussians). We can see that the lowest word error rate was obtained for the Clean/Clean and GSM/GSM baseline conditions using 32 Gaussians per density. For that case the degradation in recognition accuracy produced by GSM coding under matched conditions was about 1.0% absolute. Thus, the best absolute performance obtained from the weighted acoustic modeling method reduces the gap that exists between the best system's performance under Clean/Clean and the best performance under GSM/GSM conditions by approximately 60% (relative). The results using 64 Gaussians per density are not displayed in figure 7.3 because, as we saw in Chapter 2, there exists a data overfitting problem in this configuration and thus the error rate is greater than the 32 Gaussian configuration. As opposed to multi-style training results shown in Chapter 2, our technique shows consistent gains for the 3 configurations shown without the need of twice the amount of data when training the individual models.

If the computational cost is an issue, the optimal solution (*i.e.*, best performance in error rate terms when compared to a system with same number of Gaussians or less) arises by having two sets of 16 gaussians in a weighted model fashion. During decoding in this configuration, all 32 Gaussians per density would need to be evaluated making this a system computationally equivalent to a 32 Gaussians per density, but obtaining only slightly better results. It is thus clear that the biggest benefit of the technique proposed in this chapter can be realized at the cost of some extra decoding delay. As machines get faster, nontheless, these issues would become less critical.

The results displayed in Figure 7.3 suggest that once we have established a set of λ parameters that work with a certain number of Gaussians per state we can apply them successfully to other HMM configurations with different number of Gaussians. While we can expect equal or better results when the λs are optimized for each of the particular configurations, this involves more computation. The computational expense associated with the exhaustive weight search is mitigated by three facts: (1) the search can be performed more efficiently using model configurations with small numbers of Gaussians, (2) the phones can be clustered into categories, which will simplify the search and produce results similar to untied conditions, and (3) the search for optimal weights needs to be done only once, after the source models have been trained.

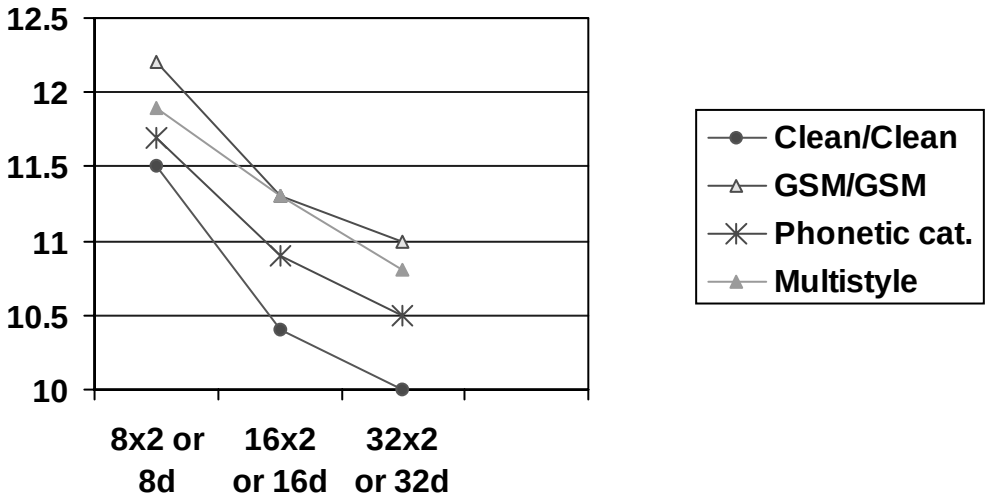

Figure 7.3 Recognition experiments using the weighted acoustic modeling, and the best set of λs under different HMM models.

7.5 Summary

In this chapter we discussed the implementation details of the weighted modeling approach based on average statistics of the relative log spectral distortion introduced in the long-term prediction. If computational costs during decoding are important, then

we can obtain slightly better results for a given decoding cost by organizing the models as in the weighted model approach proposed in this chapter. If computational cost is not considered, we observe that we can reduce the degradation introduced by GSM codec by 60 to 70 percent.

The basic premises, however, are that access to clean, uncoded data that is relevant and within the same task and domain is available to perform the model combination and that the model combination is based on models that match the distortion introduced by the codec. This last assumption makes the approach be codec dependent. In the next chapter we will explore the weighted acoustic modelling method based on instantaneous estimation of the distortion introduced by the codec, and explore the issues and challenges associated with codec-independent compensatory modeling.

Chapter 8
Weighted acoustic modeling based on instantaneous distortion estimates

In Chapter 6 we introduced the concept of weighted acoustic modeling and described the two modalities in which this approach can be implemented: based on average distortion statistics and based on instantaneous distortion statistic estimates. In Chapter 7 we further elaborated on the implementation details and presented the results obtained from the experiments using the average statistics. We observed that a reduction of up to 70% of the degradation introduced by GSM coding can be obtained using that method. These results were conditioned on the availability of undistorted data, and on the assumption that the type of codec is known and fixed.

In this chapter we explore a second approach to weighted acoustic modeling where the distortion information is based on instantaneous estimates. An approach based on instantaneous estimates of the distortion signal will be more generally applicable than an approach based on average distortion. As we saw in Chapter 7, the reason for this is the need of undistorted data and models for approaches based on average distortion. The major challenge associated with this approach is the difficulty of the estimation of the distortion introduced by the codec. However, this side information could be provided directly by the codec by computing the difference between the long-term residual and the reconstructed long-term residual. In spite of this being a simple computation that could be carried in the terminal device, up to today, no provisions have been specified for this in any major speech coding standard.

We experiment with distortion information assumed to be provided by the terminal device to determine how well we can do when such distortion information was provided to the recognizer. When this information is not available we propose two methods to estimate the codec distortion: one method is codec-dependent and is based on a second decoding pass; the second method is codec-independent and is based on long-

term predictability observed in the speech signal. Another issue of importance that we discus is associated with bounding the noise estimate or mapping it to the desired range for the model weights. The general method that we employ for this purpose is the family of piece-wise linear mappings. We finally present recognition results for the known-codec scenario and the concurrent coding scenario.

8.1 Mapping instantaneous distortion information into distortion class probability weights

When the concept of weighted acoustic modeling was introduced in Chapter 6 it was mentioned that the weighing function f in Equation 6.3 could be decomposed into the mixture coefficients $p_{q_i}(k)$ and the weighting terms $p(m|d_i)$. The weighting terms represent the probability that at frame i the observation vector a_i is emitted from distortion class (or environment) m, given that the speech frame resulted in distortion d_i in the codec block. In practice we can define a set of functions that map the value of the distortion observed into the set of weighting terms.

As we did in Chapter 7, we consider the existence of two distortion classes in order to simplify the problem. Basically, one class will model high-distortion frames and the other class will model frames with low distortion. Then, the relation between the probability of a frame belonging to each environment given the distortion d_i is defined by the expression $p(m = 2|d_i) = 1 - p(m = 1|d_i)$. In this section we describe the mapping of the distortion scalar d_i into $p(m = 2|d_i)$ assuming that Class 2 represents the distortion class.

The distortion measurement d_i is a scalar quantity that reflects the overall distortion introduced by the codec for the observation vector at frame i. Expressing this in terms of the RPE-introduced quantization (*cfr.* Equation 4.6),

$$d_i = \sum_{n \in frame} (\hat{h}[n] * \Upsilon(r[n]))^2 \tag{8.1}$$

The expression above represents the energy contained in a frame of the distortion signal introduced in the speech signal. Therefore, the range of this function is between 0 and infinity. It is also possible to base the distortion-class probability weights on the following version of this distortion quantity, which we will call “compressed” distortion:

$$\tilde{d}_i = \log(1.0 + d_i) \tag{8.2}$$

To map the degradation or compressed distortion into a range of valid probability values (*i.e.*, between 0 and 1) we will use the following family of piece-wise linear functions:

$$p(m{=}2|d_i) = \begin{cases} k_1 & d_i \le x_1 \\ \dfrac{(k_2 - k_1)}{(x_2 - x_1)} d_i + \left(k_1 - \dfrac{(k_2 - k_1)}{(x_2 - x_1)} x_1 \right) & x_1 < d_i < x_2 \\ k_2 & d_i \le x_2 \end{cases} \tag{8.3}$$

This type of function is completely defined by the points (x_1, k_1) and (x_2, k_2), with k_1 and k_2 bounded between 0 and 1. The points x_1 and x_2 are the values of the compressed distortion that serve as break-points between lines in the function. We refer to the distortion resulting from this mapping as the “bounded” distortion.

8.2 Bounds on recognition accuracy given coded-induced distortion information

Quantization or coding distortion is not normally available directly to the recognizer, and hence must be estimated. This estimate should be based on the reconstructed speech signal that the recognizer normally has access to, as well as on the knowledge of the operation and properties of the codec by which the signal was processed. In the following subsections discuss how to obtain these estimates. In this section we run experiments to determine the upper bounds on performance when the distortion information is available to the recognizer.

We have discussed in Sections 4.2 and 4.3 the effects of GSM coding distortion on the signal cepstra. The distortion signal d_i refers to the effect of the coding process on the time signal and not on the cepstral vector. As we saw in Chapter 4, this effect is interpreted as a non-stationary additive noise distortion signal in time, and it corresponds to a perturbation in the cepstral domain which is related to the time distortion signal through an intricate non-linear relation. We can reasonably expect that instead of obtaining a distortion estimate and then mapping it to the cepstral domain using Equation 4.8, it would be simpler and more useful to compute the cepstral features at the terminal device and compare directly these cepstral vectors $\mathbf{s}$ and $\hat{\mathbf{s}}$, transmitting information related to the level of the cepstral distortion observed. Let $\mathbf{s}_i[j]$ denote the j^{th} coefficient of the i^{th} cepstral vector of the clean signal and $\hat{\mathbf{s}}_i[j]$ the corresponding cepstral coefficient of the cepstral vector derived from the coded speech. We can define the normalized distance between the two 13-dimensional cepstral vectors as

$$\delta_i = \frac{1}{12}\sum_{j=1}^{12}\left(\frac{(\mathbf{s}_i[j]-\hat{\mathbf{s}}_i[j])}{\mathbf{s}_i[j]}\right)^2 \tag{8.4}$$

The above equation does not include the zeroth cepstral component *(i.e.*, the frame energy). This expression will reflect the normalized average distortion introduced to the cepstral vector due to coding. It is necessary to point out here that this is a front-end dependent feature. Another interesting fact is that the various cepstral coefficients can be weighted differently depending on how much these coefficients might influence recognition.

The above quantity could then be calculated for each frame and mapped into the weighting probabilities through the use of the type of piece-wise linear equations described in Equation 8.3 and the result presented to the recognizer in the way illustrated in Figure 8.1.

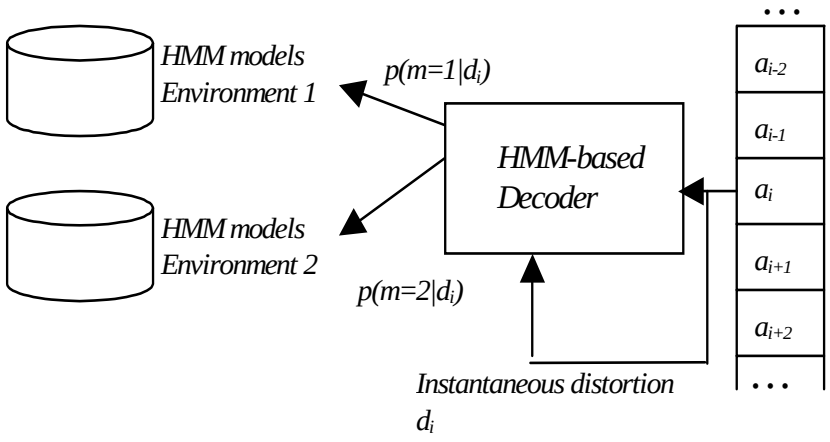

Figure 8.1 Block diagram of a weighted acoustic modeling based decoding configuration in which oracle cepstral distortion information is provided to the decoder.

We performed recognition experiments using the following setup: for acoustic models, we used the flat-weight system described in Section 7.4, combining the models trained on GSM and trained on clean speech. During decoding, the cepstral distortion quantity described above was computed for each frame, mapped into a bounded distortion quantity through a 3-line mapping (as described in Section 8.1) and passed to the recognizer. The recognizer weighted the mixtures using this bounded cepstral distortion information. In other words, in this experiment we adjusted those mixture weights

of the model according to a measurement of the cepstral distortion rather than blindly adjusting them based on phonetic identities of the senones. The parameters of the linear mapping were established by maximizing the accuracy of the test set.

Figure 8.2 shows the results obtained using the setup described above as a function of the number of Gaussians per HMM state. The labeling of the horizontal axis denotes the number of Gaussians per state for the weighted acoustic modeling scheme (top row of the horizontal axis' labels) and the number of Gaussians per state for the source models in the baseline conditions (bottom row of the horizontal axis' labels).

We can see in Figure 8.2 that overall results for each number of Gaussians (except 64x2 or 64) were improved by considering instantaneous distortion estimates, compared with the "Phonetic categories" scheme which employed average phonetic distortion information. The best relative improvement occurs at 8x2 or 8d Gaussians per mixture and gives a result almost as good as the Clean/Clean conditions. The best absolute results occur also at 32x2 or 32d (which is also the best point of the curve for Clean/Clean conditions and matched GSM/GSM conditions. In this case the gap introduced by GSM coding gets reduced by 70%: this gap goes from 1.0% absolute to 0.3% absolute.

Evidently, these results are based not only on the assumptions of availability of clean uncoded relevant speech (which was also an assumption of the methods using weighted acoustic modeling based on average distortion), but also on the availability of instantaneous distortion estimates that substitute the lambdas that were based on phonetic statistics. Intuitively, we expect that the resulting performance of a system that is based on distortion estimates (as opposed to cepstral distortion information provided by the terminal device) and that was trained using only GSM-coded data (as opposed to clean and GSM models) will be bounded by these results based on best-case scenarios. The experiment described above bounds the performance of realistic scenarios.

We can see that the recognition performance benefits from the instantaneous cepstral distortion information. It would be a very simple and inexpensive operation for the codec and the channel to transmit, for every frame, the precomputed scalar described in Equation 8.4. Front-end dependencies aside, this terminal-derived information would benefit the performance of the ASR application. However, because current standards don not implement this operation, we propose methods to estimate the instantaneous distortion information in the following sections.

Another difference between the experiment described here and the recognition experiments performed in the remainder of the chapter is related to the issue of model retraining. Because the experiments described here involve the combinations of clean and GSM models we are assuming that there are clean uncoded data available as well as coded data. This data does not have to come from a simultaneous recording under different channel conditions (*i.e.*, "stereo" recording conditions). As we mentioned in the Chapter 7, this might not be the case: sometimes the availability of within-domain, relevant, uncoded data is limited or non-existent. Therefore, we will employ the method of estimation of the instantaneous distortion as a method to weight the contribution of each frame to each of the distortion environments probabilistically. Our distortion estimates will be noisy versions of the actual distortion information. Because of these problems, we should expect the actual results obtained with estimated distortion information to be worse than the upper bound established in this section,

8.3 Instantaneous distortion estimation based on recoding sensitivity

In this section we derive an estimate of the instantaneous distortion introduced to the speech signal by means of a second GSM coding pass on the GSM-coded speech. The second GSM coding pass is introduced in order to perform a comparison between the received speech (*i.e.*, what would constitute the input to the second decoding pass) and the speech resulting from the second decoding pass. It is important to stress the

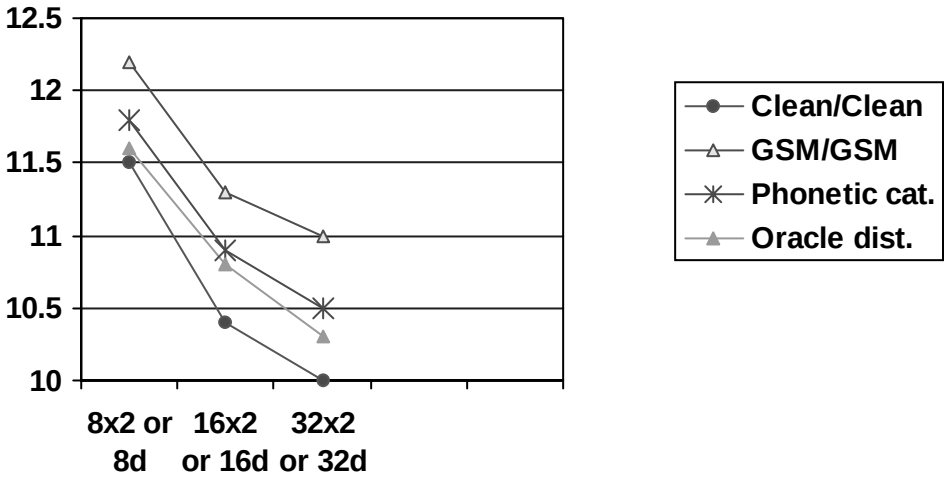

Figure 8.2 Recognition results employing instantaneous cepstral distortion information provided by the terminal device and flat-weight acoustic modeling as a function of the number of Gaussians per HMM state.

fact that the received speech will be employed for recognition: the second GSM coding is only applied in order to determine the estimate of the distortion weights. We now describe the motivation for this approximation.

Let $\hat{s}_1[n]$ be the quantized signal, *i.e.* the signal coming out of the GSM communication link. We established in Chapter 4 the relation between the quantized speech signal and the original speech signal. This relation is given by:

$$\hat{s_1}[n] = s[n] + \hat{h_1}[n] * \Upsilon(r_1[n]) \tag{8.5}$$

At the recognizer's end of the GSM communication link we only have access to the quantized speech signal, not to the original signal. Let us assume that in the recognizer's end of the communication link, this quantized speech signal is passed through a second GSM codec. Let the output of this second decoding pass be signal $\hat{s}_2[n]$. We can then express this signal in terms of the original speech signal $s[n]$, the quantized

LPC synthesis filters of the first and second GSM coding passes $\hat{h}_1[n]$ and $\hat{h}_2[n]$, and the distortion functions introduced by the RPE representations of the long-term residual of the first coding pass and the second coding pass $\Upsilon(r_1[n])$ and $\Upsilon(r_2[n])$:

$$\begin{aligned} \hat{s}_2[n] &= \hat{s}_1[n] + \hat{h}_2[n] * \Upsilon(r_2[n]) \\ &= s[n] + \hat{h}_1[n] * \Upsilon(r_1[n]) + \hat{h}_2[n] * \Upsilon(r_2[n]) \end{aligned} \tag{8.6}$$

Equation 8.6 above means that the output of a second decoding pass will be equal to the original signal, plus the quantization introduced by the first decoding pass, plus a quantization introduced by a second decoding pass. The quantization introduced in the second decoding pass is a function of the quantized LPC filter computed in the second GSM pass, and the long-term residual signal observed in the second GSM decoding pass.

We now use Equations 4.1 and 4.2 to derive an expression relating the long-term residual with the short-term residual signal and the estimated long-term residual signal, and then rewrite this in terms of the original speech signal $s[n]$. Specifically,

$$\begin{aligned} r[n] &= e[n] - \bar{e}[n] \\ &= \hat{h}^{-1}[n] * s[n] - \bar{e}[n] \end{aligned} \tag{8.7}$$

Where $\hat{h}^{-1}[n]$ is the quantized LPC *analysis* filter.

We now rewrite the overall GSM quantization noise introduced in the second decoding pass $\Upsilon(r_2[n])$ making use of the result in Equation 8.7 above:

$$\begin{aligned}\Upsilon(r_2[n]) &= \Upsilon(\hat{h}_2^{-1}[n]*\hat{s}_1[n] - \overline{e_2}[n]\ ) \\ &= \Upsilon(\hat{h}_2^{-1}[n]*(s[n] + \hat{h}_1[n]*\Upsilon(r_1[n])) - \overline{e_2}[n]\ )\end{aligned} \tag{8.8}$$

Expanding the terms inside the parentheses:

$$\Upsilon(r_2[n]) = \Upsilon\left(\hat{h}_2^{-1}[n]*s[n] + \hat{h}_2^{-1}\hat{[n]}*\hat{h}_1[n]*\Upsilon(r_1[n]) - \overline{e_2}[n]\ \right) \tag{8.9}$$

The above expression reflects the relation that exists between the quantization introduced during the second GSM coding pass in terms of the original signal, the quantization in the first decoding pass, and the long-term residual estimate of the second decoding pass.

We make two basic approximations. The first approximation is related to the LPC analysis performed in the GSM codec. We assume that the quantized LPC vector of the first GSM pass and the second decoding passes are very similar. This is not strictly true because the second coding's LPC estimate will be based on an autocorrelation function of the speech signal plus the quantization noise introduced in the first coding pass. If the GSM quantization noise introduced in the first codec pass is not abnormally large, the LPC-based spectrum estimation process of the second coding pass will then be dominated by the spectrum of the original speech signal. The second approximation we make is that the long-term predictability structure of the speech signal is preserved after the first GSM coding pass. We can summarize these two assumptions as,

$$\begin{aligned}\hat{h}_2^{-1}[n] &\approx \hat{h}_1^{-1}[n] \\ \overline{e_2}[n] &\approx \overline{e_1}[n]\end{aligned} \tag{8.10}$$

Substituting the corresponding signals in equation 8.9, we obtain:

$$\begin{aligned}\Upsilon(r_2[n]) &\approx \Upsilon\left(\hat{h}_1^{-1}[n]*s[n] + \hat{h}_1^{-1}\hat{}[n]*\hat{h}_1[n]*\Upsilon(r_1[n]) - \overline{e_1}[n]\right) \\ &\approx \Upsilon(\hat{h}_1^{-1}[n]*s[n] - \overline{e_1}[n] + \Upsilon(r_1[n])) \\ &\approx \Upsilon(r_1[n] - \Upsilon(r_1[n]))\end{aligned} \tag{8.11}$$

The expression above implies that with the two approximations summarized by Equation 8.10, the quantization introduced in the second decoding pass depends on the long-term residual of the first coding pass and the quantization introduced in the first coding pass.

If we further assume that the first-pass long-term residual is substantially larger in energy terms than the quantization introduced to it by the RPE codec, then the expression above can be rewritten as:

$$\Upsilon(r_2[n]) \approx \Upsilon(r_1[n]) \qquad \text{if in energy terms} \quad r_1[n] \gg \Upsilon(r_1[n]) \tag{8.12}$$

In the next section we illustrate how well the long-term predictability assumption holds in real conditions.

We now incorporate the above conclusions into the context of the proposed weighted acoustic modeling method. Based on the analysis developed above we propose to employ a second coding pass on the coded speech coming out of the GSM communication link and perform a comparison between the output of this second coding pass and the observed first-pass coded speech. This gives us directly the second-pass quantization signal $\Upsilon(r_2[n])$ which we use as an estimate of $\Upsilon(r_1[n])$. Using this estimate of the distortion introduced in the first GSM pass we then map the estimated distortion into the distortion class weights using a piece-wise linear function of the type described in the previous subsection. Figure 8.3 illustrates this process.

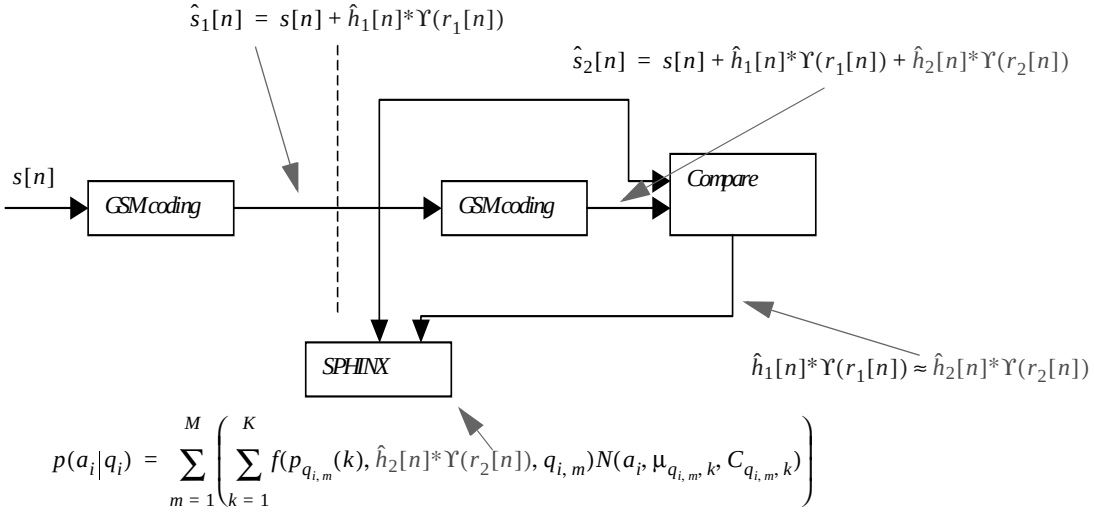

Figure 8.3 Block diagram of the process proposed to compute an estimate of the instantaneous distortion introduced by the GSM coding pass by means of a second coding pass.

It is important to stress that this method is codec dependent, *i.e.*, we not only need to know the type of coding algorithm that affected the speech signal, but we would also need to have an implemented replica of the coding program or algorithm in order to perform the second coding pass. In the next section we describe a method that is codec-independent.

As discussed above, the distortion estimate based on the second decoding pass depends on several assumptions. We now assess the quality of these approximations using off-line simulations. We performed these simulations by processing a segment of the TIMIT database using the GSM codec, and then reprocessing the output of the first coding pass. In order to establish the quality or similarity between the distortion introduced in these subsequent coding passes we computed the following two types of distortions: (1) the distortion between the original speech signal and the output of the first decoding pass, and (2) the distortion between the outcomes of the first and second

decoding passes. Based on these two measured distortion signals, we computed the energy of subframe-length segments of these two different signals and kept them as paired points. Figure 8.4 below shows the scatter plot of these sets of paired measurements for a subset of TIMIT regardless of phonetic identity and including silence regions.

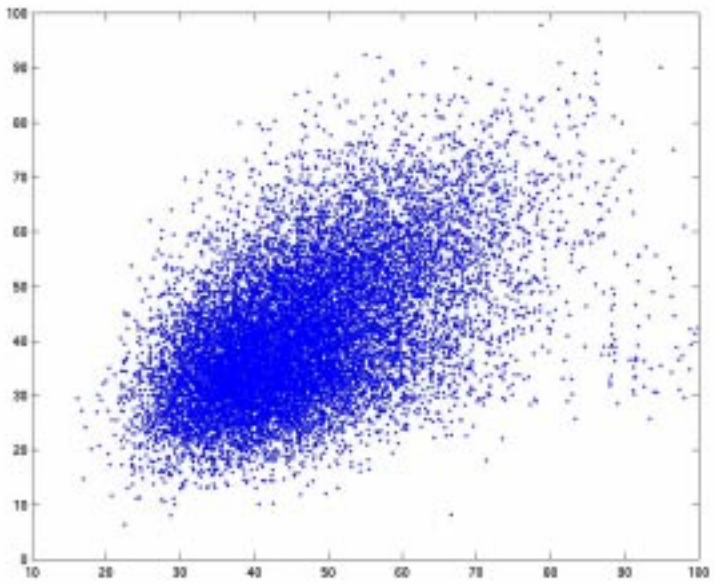

Figure 8.4 Scatter plot of distortion introduced in the first GSM coding pass (vertical axis) versus distortion introduced in second GSM coding pass (horizontal axis).

From Figure 8.4 above we can get a sense of the goodness of the approximation introduced in Equations 8.10. The correlation between these pairs of points is 0.55. It is certainly not a very strongly correlated relation, and it would be hard to identify the individual quality of each of the assumptions made. However, one can imagine that for different phone identities, the validity of the assumptions will hold in different manners. For example, the strong periodicity found in vowels might result in better correlation between the distortion of the first and second decoding pass.

In order to verify this idea, we recomputed the scatter plot but in this time we only kept the data associated with the vowel **eh**. In this case, as we can see from figure 8.5,

the relation existing is slightly better. The corresponding correlation coefficient for this case is 0.6.

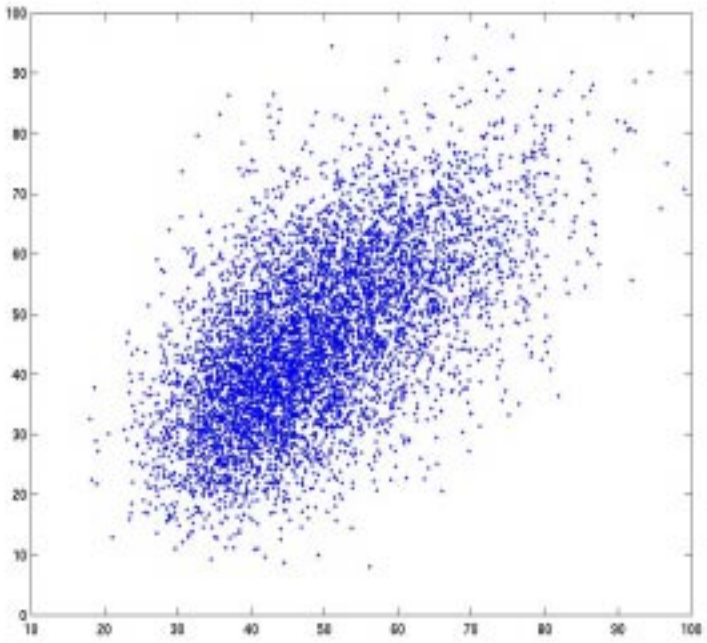

Figure 8.5 Scatter plot of distortion introduced in the first GSM coding pass (vertical axis) versus distortion introduced in second GSM coding pass (horizontal axis) for the realizations of the phoneme **eh**.

Table 8.1 shows the results of recognition experiments performed when the re-coding sensitivity was used to estimate the instantaneous distortion. Unlike the experiments described in Chapter 7 and in the previous subsection, this experiment involved the retraining of the acoustic models. This information was used to evaluate the probability of state emissions both in training and in Viterbi decoding using Equation 6.7. We can observe that the recognition results we have in the bottom row go beyond the baseline Clean/Clean results obtained with 8 Gaussians per density. This is an indication that the process of retraining considers all the Gaussian densities in the mixture and retrains them as a single set of $nx2$ Gaussian densities. In contrast, the experiments in Chapter 7 were performed by training these models with n Gaussians separately and afterwards, during decoding, treating them as the interpolation of 2 surfaces. In the extreme case that we assign flat or constant weights to each frame (*e.g.*, 0.5) then this

procedure will be algorithmically identical to standard Baum-Welch retraining and Viterbi decoding except that we would be using twice as many Gaussians as in the original source models. In the case that we assign random weights, the equivalent resulting model after retraining will be equivalent to a model with as many Gaussians as the number of Gaussians in any one of the conforming source models. Further analysis and experiments will be presented in subsequent sections.

Test data	Training data	% WER
Clean	Clean	**11.5%**
GSM	Clean	13.0%
GSM	GSM	**12.2%**
GSM	GSM+Clean (multistyle)	11.9%
GSM	2 d.c. 15 phone-class optimal weights	11.7%
GSM	GSM (re-coding sensitivity based multimodeling)	**11.0%**

Table 8.1 Recognition results of baseline results and experiments employing instantaneous distortion estimates based on recoding sensitivity. All the models had 8 Gaussian densities per mixture.

8.4 Instantaneous distortion estimation based on long-term predictability

We have described how using a second speech coding pass we can obtain some estimate of the distortion introduced in the first decoding pass. We also said that this information is codec dependent. In order to move towards a codec-independent approach in this section we will focus on properties of the speech signal itself.

In Sections 3.3 and 3.5 we mentioned that the quality of the coding of the long-term residual is related to the quality of the prediction done by the long-term predictor block (*i.e.*, the LTP block for the case of GSM, and the adaptive codebook in the case of CELP). Thus, if a signal is highly periodic and presents a strong long-term predictability, the long-term residual (or unpredictable portion of the signal) will have small energy and the quantization introduced by the long-term residual coding will be small. In general, as long as the short-term residual codec operates on the basis of the long-

term predictability, this property will be true independent of the implementation of the long-term quantizer. Thus we should focus on the long-term predictability of the speech signal as a source of the codec-introduced distortion estimate.

During the short-term residual coding process, the long-term prediction gets computed in terms of an adaptive codebook or in terms of the long-term prediction parameters, for CELP and GSM type of coding, respectively. In the most general abstraction of these algorithms, a lag or adaptive codebook index is computed, and a gain is estimated. The lag or adaptive codebook index tries to track the pitch of the speech signal for voiced segments. The gain can be interpreted as a measurement of the long-term predictability of the short-term residual [51]. We propose a variation of this metric, which we will refer to as long-term predictability metric or LTPM, based on the logarithm of the ratio between the highest value of the cross correlation between the current subframe and the N adjacent subframes divided by the energy of the current subframe:

$$LTPM[n] = \log\left(\frac{\max_{0<p<TN}\left(\sum_k s[n]s[n+k+p+N]\right)}{\sum_k s[n+k]s[n+k]}\right) \tag{8.13}$$

The most salient issues to consider in the equation above are determining what the length of the subframes should be, and how many adjacent subframes should be considered. The actual values of these parameters can change from codec to codec, but in general, this variability of parameter values should be small for comparable types of codecs (such as: low delay coding, very low bit rate coding etc.)

In order to analyze the potential usefulness of the LTPM in estimating the distortion introduced to the signal by a codec, we explore in the next subsections two important relationships. We first consider the extent to which the LTPM is preserved after coding. We then consider the relation between the LTPM of the speech signal and the

cepstral distortion introduced. The first relation, the preservation of the LTPM across GSM-coding, is relevant also for re-coding sensitivity based distortion estimates; this was one of the assumptions represented in Equation 8.10.

8.4.1 Preservation of the LTPM across GSM coding passes

In order to obtain distortion estimates based on the LTPM of the original speech signal we will need to have an estimate of it based on the observed *coded* speech signal. We can expect these long-term predictability properties of the speech signal to be preserved as this information is usually represented in terms of a lag and a gain parameter (or an index to an adaptive codebook and a gain). These parameters, within a certain quantization error, will exactly produce a signal with the same level of periodicity as that of the observed original signal. The basic assumption, however, is that the quantization noise introduced to the reconstructed long-term residual is uncorrelated with the long-term residual signal itself. An uncorrelated additive term will have small influence the second time Equation 8.13 gets computed (*i.e.*, in the observed coded speech signal). Figure 8.6 below shows the measured LTPM for an utterance of the TIMIT database at different coding stages. The panel on the top is the LTPM for the original signal, the panel in the middle shows the LTPM for the signal after it has undergone one GSM coding pass, and the panel in the bottom for the signal after two GSM coding passes. We can see that the general contour of LTPM is preserved in general regardless of the coding passes. The correlation coefficient between the LTPM of the original speech signal and the LTPM of the outcome of the first GSM pass is 0.91. The correlation coefficient between the LTPM of the first GSM and the LTPM of the second coding passes is 0.93.

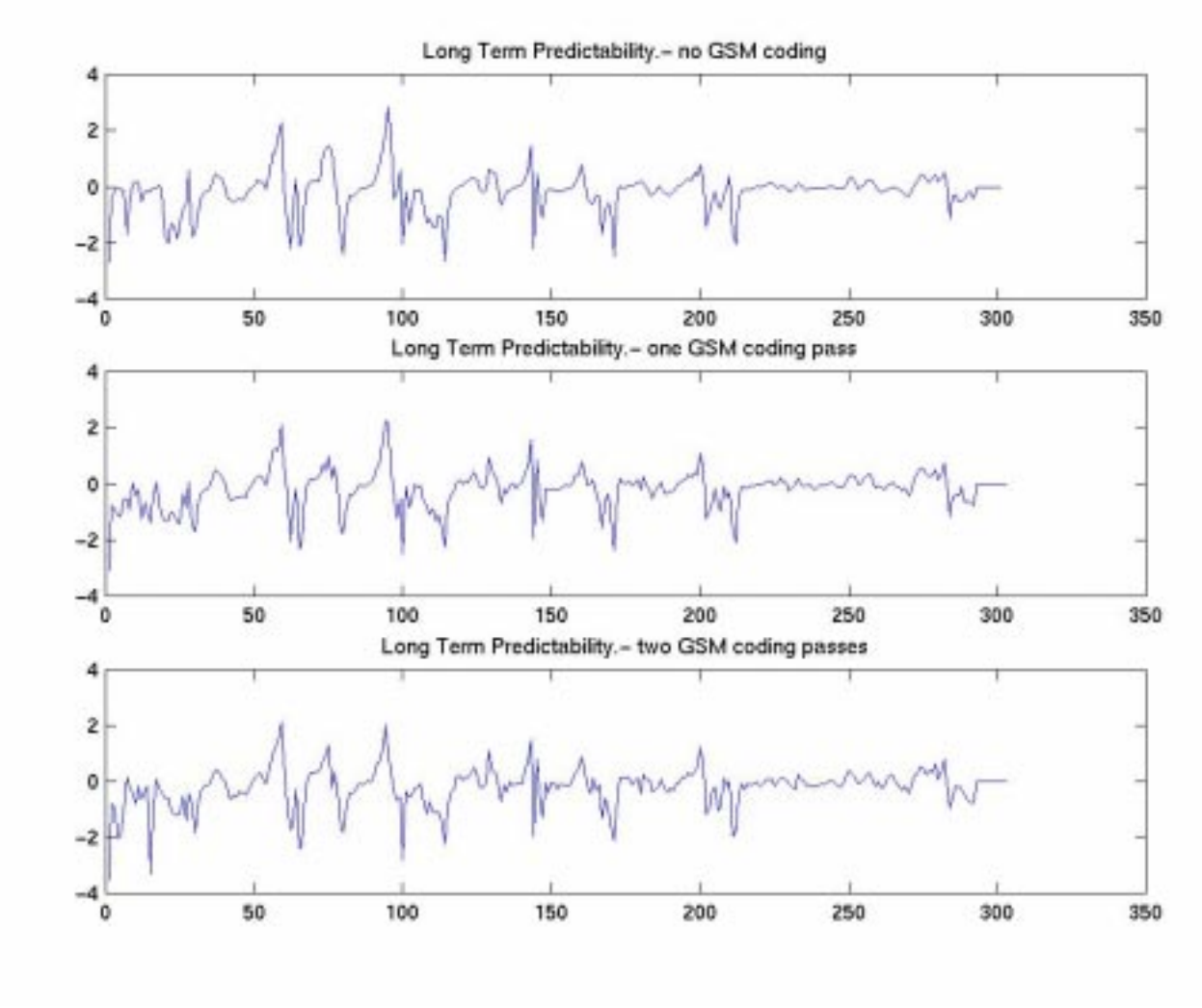

Figure 8.6 long-term Predictability for a TIMIT utterance of the original signal (top panel), after one GSM coding pass (middle panel), and after two GSM coding passes (bottom panel).

8.4.2 Relation between LTPM and cepstral distortion

In Section 8.2 we illustrated the utility of instantaneous distortion by performing an experiment based on cepstral distortion provided to the recognizer by the terminal device. Both the second coding pass method and the LTPM based method provides us with estimate of the temporal distortion signal and a metric of predictability, but neither of these are defined directly in the cepstral domain. Figure 8.7 below shows the plot of the LTPM for a certain utterance of the TIMIT corpus as defined in equation

8.13 and the plot for the normalized cepstral distortion as described in equation 8.4 for the same utterance. We can see that the relation is not as clear as the one that exists between the LTPM plots above. We believe that this is due to the non-linearities that in the relations between the LTP information and the normalized cepstral distortion. The correlation coefficient for these two signals is less than 0.1, a low value seemingly due to their non-linear relationship. In the following recognition experiment section, we evaluate the impact of employing LTPM in recognition in spite of the observed low correlation to cepstral distortion.

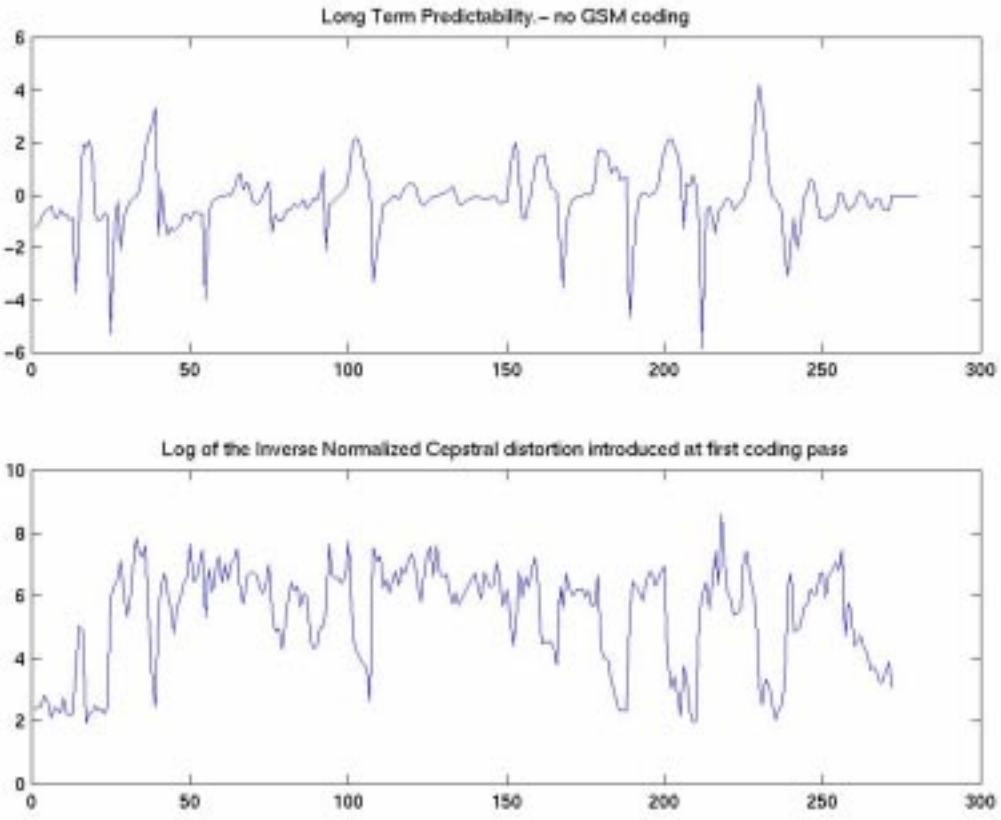

Figure 8.7 long-term predictability (top panel) for a TIMIT utterance, and corresponding cepstral distortion introduced by coding (bottom panel).

8.5 Recognition experiments: weighted acoustic modeling under GSM coding using instantaneous distortion information

In this section we describe some experimental results obtained when we applied the instantaneous distortion estimates to the weighted acoustic modeling method. We first describe experiments performed using the TIMIT corpus, and subsequently the experiments on the real Telefónica TI+D GSM database.

8.5.1 TIMIT experiments

We performed experiments using the TIMIT corpus and a system similar to the system of section 7.4. The main difference between the systems was the use of instantaneous distortion instead of the average distortion. A modified version of the SPHINX-3 trainer and decoder were employed during these experiments. These modified system was altered so it was able to accept files containing distortion information parallel to the cepstral features, and it provides with access to these instantaneous distortion data to the modules that compute the Likelihoods.

Figure 8.8 below shows the recognition results obtained using recoding sensitivity as a distortion estimate. In every case, the system models were initialized from standard GSM/GSM models and based on two distortion categories retrained using the provided distortion information. This retraining process actually consisted of a few passes (typically 5) of the Baum-Welch algorithm.

The comparison is made with respect to the number of Gaussians: the baseline system with n is compared against the weighted acoustic modeling based on $\left(\frac{n}{2}\right)$x2 number of Gaussians. For example, the Clean/Clean condition with 16 Gaussians should be compared with the system using two 8-Gaussian weighted models (labeled as 8x2). The

rationale for this, as we explained at the of section 8.3, is that this is essentially a Baum-Welch retraining process.

For the three different numbers of Gaussians tested, the recoding-based system provides only small improvement in accuracy. This limited results can be contrasted with the results obtained using cepstral distortion information provided by the terminal device in Section 8.2. It is apparent that the usefulness of the second decoding pass as estimate of cepstral distortion is limited. The best results are obtained using a 16x2 weighted modeling system. However, almost identical results are obtained using a system with half the number of Gaussians based on the coding-sensitivity method: 8x2 weighted modeling conditions. We can achieve close to best performance under GSM coding using the weighted acoustic modeling approach with only half the number of Gaussians that conventional acoustic modeling needs.

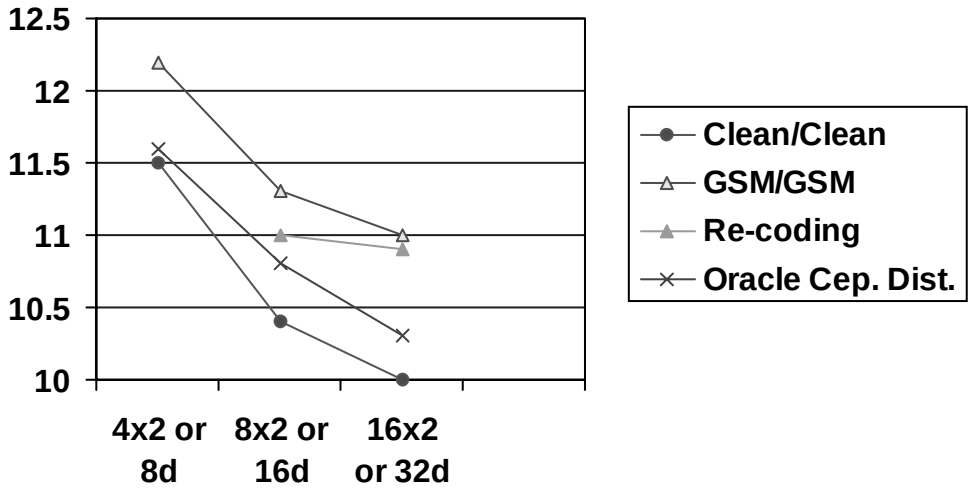

Figure 8.8 Recognition results using instantaneous distortion information based on recoding sensitivity as a function of number of Gaussians per state on GSM coded speech.

Figure 8.9 shows the results obtained when the acoustic modeling approach is based on LTPM information. In this case, the best results are obtained with a 16x2 system, which has the same number of Gaussian components that the 32d system, therefore no computational advantage is gained. The best results are slightly better than

those obtained using recoding sensitivity information: 0.2% absolute, or a 20% reduction in the performance gap introduced by GSM codec. In both cases the best results are obtained using the 16x2 Gaussians per state configuration.

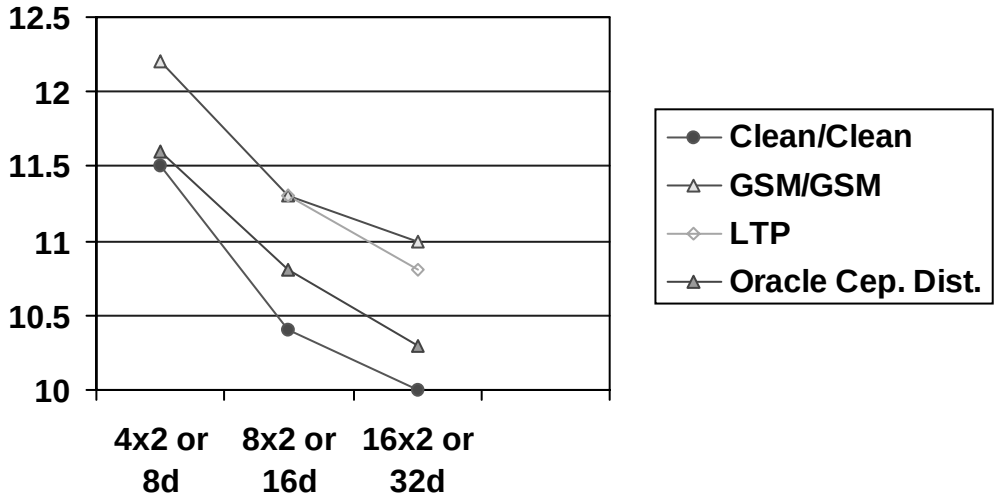

Figure 8.9 Recognition results using instantaneous distortion information based on long-term predictability as a function of number of Gaussians per state on GSM coded speech.

We have seen that there is a small but consistent gain when long-term predictability information is used in recognition: recognition seems to benefit from this type of information. Historically, in speech recognition, the front-end has been designed to ignore long-term correlations thus making the ASR free of LTP information. We now would like to evaluate the effect of reorganizing our approach to ASR using LTP information even when no speech coding is present. In other words, we would like to see if the gains observed in the experiment above are due to the fact that LTP information adds useful information to the classification process or if instead if these gains are actually obtained because the LTP information provides information that is useful only in the context of speech coding. An example of such a situation is information that combined with the weighted acoustic modeling approach help us in “organizing” or structuring our data based on distortion categories. If this is true, then the proposed approach is relevant only when there is a speech coding process affecting the signal.

In order to explore the above question, we performed the following set of recognition experiments. Figure 8.10 below shows the results of such experiments when GSM coding is not affecting the signal. The LTP information has been included in the same way as the experiment above when GSM is present. In this case the two distortion models were both initialized by the clean standard HMM models. Afterwards, the same number of Baum-Welch iterations as in the last section experiment's were performed. We tested two configuration cases, the performance actually degraded slightly but consistently.

Based on the outcomes of this experiment, we can conclude that LTP information is useful to help in weighted environments when a distortion is affecting the signal. When no distortion is affecting the signal, the LTP information is of no help. Current front-ends that make no use of long-term correlations will not provide the recognizer with additional useful information if they are made to model long-term predictability. The value of the long-term information is that it provides ways to organize and weight the contribution of each distortion environment according to the distortion introduced to the signal through coding. Furthermore, the fact that the performance achieved is actually inferior to the equivalent system with twice the number of Gaussians might be an indication that perhaps the weighted acoustic method applied to TIMIT presents some problems which could be improved (for example, better mapping from distortion information estimate to model weights).

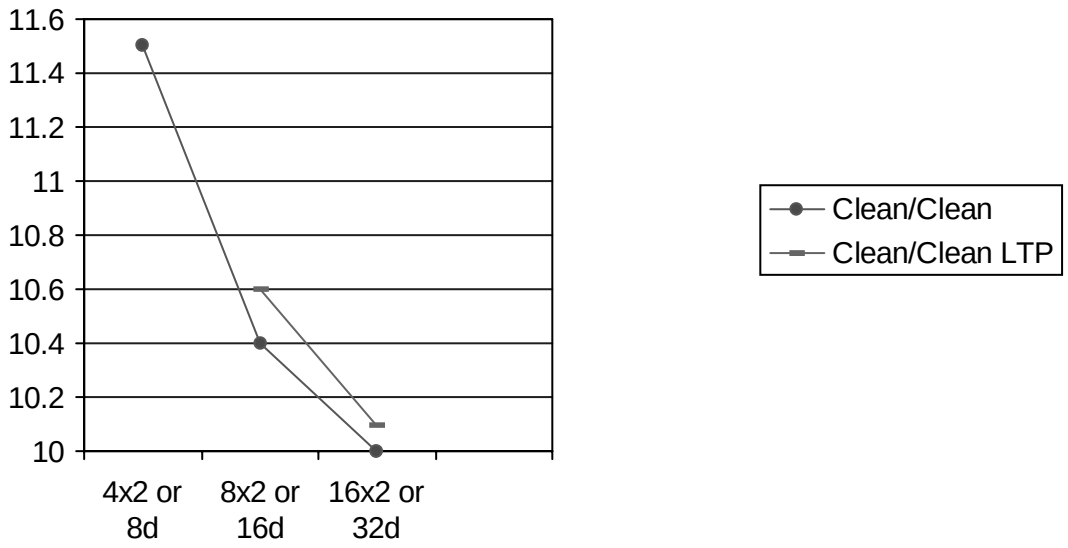

Figure 8.10 Recognition results using instantaneous distortion information based on long-term predictability as a function of number of Gaussians per state on clean uncoded speech.

8.5.2 Experiments using Telefónica (TID) database

We performed experiments based on the Telefónica database in order to evaluate the performance of the proposed methods on systems trained using data coming from a real GSM environment. The basic system was trained using the entire data in the training set of the TID corpus, and the two environmental categories were initialized using these models. The weights to scale the environments were then calculated based on instantaneous distortion derived from the recoding sensitivity of the observed data, for the first experiment, and derived from the LTPM information, for the second experiment. The mappings from these types of measurements to the model weights were performed using the same mappings derived for the TIMIT corpus: no further fine-tuning of these functions was performed. Results of recognition experiments are shown in Table 8.2. In absolute terms the error rate reduction achieved is 0.3% and 0.1%. It is not possible to determine, in this case, the reduction of the performance gap introduced by the GSM codec achieved by the proposed methods, as the TID database includes no

GSM uncoded "stereo" information (*i.e.*, a set of parallel clean utterances). We can only refer to absolute and relative reduction of the overall error rate.

We should also mention that in this corpus, the GSM coding is a homogeneous phenomenon but is not the only source of acoustic degradation. All the files in this corpus were homogeneously received from a similar GSM communication link, however the type and intensity of the noise, the dialectal differences of the speakers and other sources of variability etc. vary radically from speaker to speaker. We consider it remarkable that in spite of these acoustic variabilities, we observed an improvement in performance on this database when we applied the LTP and the recoding sensitivity based methods aimed at minimizing the effect of GSM..

	WER%
Baseline	6.9%
Recoding sensitivity	6.6%
LTP	6.8%

Table 8.2 Recognition experiments for the Telefónica database employing the two proposed methods of weighted acoustic modeling.

8.6 Speech recognition under concurrent speech coding conditions

The technique proposed in Section 8.3 makes use of a second decoding pass to produce an estimate of the distortion introduced on the first decoding pass. Naturally, the biggest constraint of this technique is related to the assumption that the recognizer only receives data from the GSM network. In practical conditions, this assumption would only be valid in places where there is only one digital cellular telephony standard (*e.g.*, Europe). In the U.S. several cellular networks coexist, thus limiting the applicability of the proposed technique. An alternative for the recognizer is to attempt to classify the type of codec that affected the signal and produce a hard decision. Associated to this approach is the problem of doing the estimation and the problems associated with estimating the wrong environment.

In order to circumvent the codec dependency we proposed in section 8.4 to base the distortion estimate on LTPM information. This means that we moved away from estimating the instantaneous distortion using knowledge about operational properties of a specific codec, to estimating this distortion using knowledge about general operation properties of a *family* of codecs as well as properties of the speech signal itself. We argued that if a codec performs the compression of the speech signal based on the short-term and long-term analysis of the speech, then the distortion introduced to the speech signal due to long-term residual quantization will be determined by the properties of signal itself. In other words, the quantization mechanism that introduces distortion to the reconstructed long-term residual will be limited by the quality of the long-term prediction block, which is constrained by the predictability of the signal.

In this section we present the experiments we performed using data that has been coded by either the GSM or the FS-1016 codecs with equal probability but the type of coder is unknown to the recognizer.

8.6.1 Baseline experiment: multistyle training as an alternative solution

Figure 8.11 shows the results of the set of baseline recognition experiments for this section. In addition to the Clean/Clean and GSM/GSM curves that we have been displaying previously we can observe 3 new conditions: CELP/Clean, CELP/CELP and Conc/Conc. The conditions CELP/Clean and CELP/CELP refer to the case where the testing data undergoes a FS-1016 CELP coding process while the *training* data is Clean or CELP coded respectively. The condition Conc/Conc refers to matched concurrent conditions, meaning that any utterance file in the training set and the testing set has 50% chances of being coded by either the GSM codec or the CELP codec. The Conc/Conc conditions are equivalent to a *multistyle* system: utterances coming from both coding conditions are mixed in the training data. There is no attempt to classify the data during training nor recognition.

Performance wise, we can see that the curve of the CELP/Clean conditions is substantially worse than the best case condition (*i.e.*, the Clean/Clean scenario). When matched models are used to decode the CELP data the gap introduced by the codec is cut approximately by half. The matched concurrent coding condition (Conc/Conc) curve falls in between the matched GSM/GSM and CELP/CELP models, being just slightly closer to the CELP/CELP curve.

Overall, the relative increase in error rate introduced by the Concurrent coding with matched model conditions (*i.e.*, the multistyle scenario) is approximately 20%. For example the WER goes from 10% to almost 12% when 32 Gaussian densities per state are used (this is also the case of the best number of Gaussians per state).

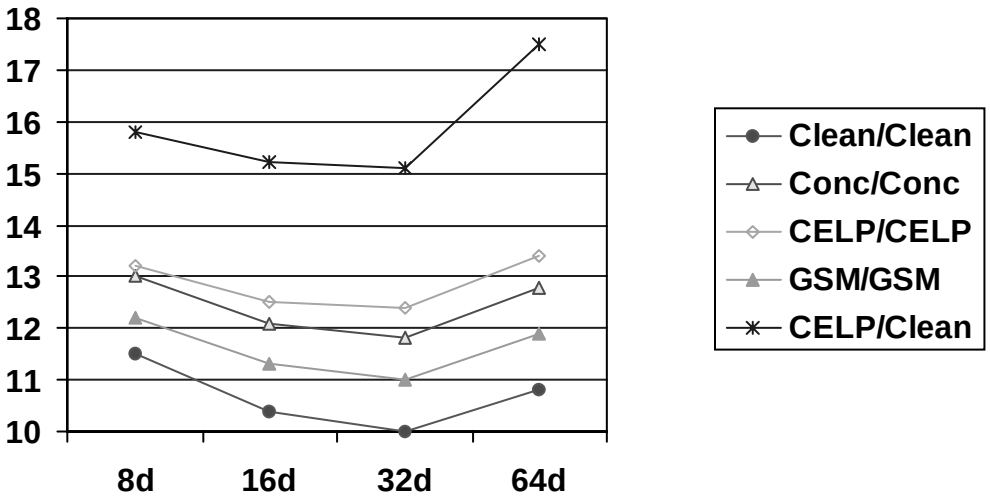

Figure 8.11 Word Error rate for baseline recognition experiments: Clean, GSM, Concurrent and CELP conditions.

8.6.2 Concurrent coding recognition experiments using cepstral distortion information

The same way that Section 8.2 established a performance upper bound curve for the case of GSM coded speech using cepstral distortion information provided by the terminal, in this section we will employ similar cepstral distortion information to establish this upper bound curve for the case of concurrent coding. The cepstral distortion experiments will be performed under two scenarios; both scenarios are based on ceptral distortion information provided externally to the recognizer. In the first scenario we perform weighted acoustic modeling using clean models and concurrent (or multistyle) models. In the second scenario, which we refer to as structured modeling, we perform weighted acoustic modeling keeping the coding environments separate. In neither of this cases we will retrain our models; instead, we assume that undistorted data is available. In future sections compare these results with situations in which oracle information and undistorted models are not available, and models have to be retrained.

8.6.2.1 Recognition experiments using cepstral distortion information without model retrain

We performed experiments similar to those in Section 8.2, but in this case the GSM codec was replaced by a concurrent GSM/CELP coding scenario. The cepstral distortion introduced by the coding process was provided as oracle information to the decoder. The weighted acoustic modeling was performed using a set of models trained from clean speech and another set of models trained from concurrent GSM/CELP coded speech, *i.e.*, multi-style modeling as described in Section 8.6.1. In this case no retraining of the acoustic models was performed. The results obtained are shown in Figure 8.12.

The best results are obtained with 32 Gaussians per mixture. In this case the error gap between the Clean/Clean and the Conc/Conc scenarios is reduced by almost 40%. These benefits are obtained using oracle information and clean data models, thus the 40% reduction of the degradation gap is a best case upper bound. As was the case with the GSM-only experiments in the previous subsections, we expect the non-oracle experiment results to be bounded by the oracle experiments' results once the oracle distortion data is replaced by estimates of the distortion, and the assumption of availability of clean data models is removed.

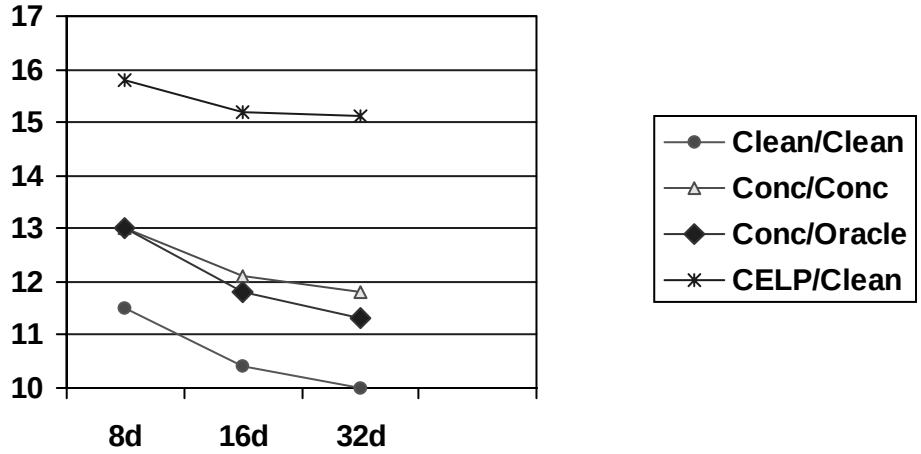

Figure 8.12 Recognition results on concurrent coding and using oracle cepstral distortion information.

8.6.2.2 Recognition experiments using oracle cepstral distortion information and structured model

The results obtained in the previous subsection are based on the assumption that the long-term coding process of the residual signal affects the GSM and CELP coded speech similarly. Once a level of distortion has been established for a given frame, the frame's likelihood is evaluated using a weighted acoustic modeling based on that distortion level information only: *i.e.* we quantify the codec distortion, but we do not attempt to qualify it. The acoustic model representing the distorted environment in the previous section's experiment reflects the effect of concurrent coding without being specific about the type of codec that each utterance encountered. In order to realize the maximum possible benefit from the oracle information, we could also further structure the acoustic model organization in order to reflect the type of coding applied to the speech signal. We refer to this model organization as structured weighted acoustic modeling. In this case, the model interpolation is performed between clean models and matching codec conditions models, for each of the coexisting codecs. This process is illustrated in figure 8.13.

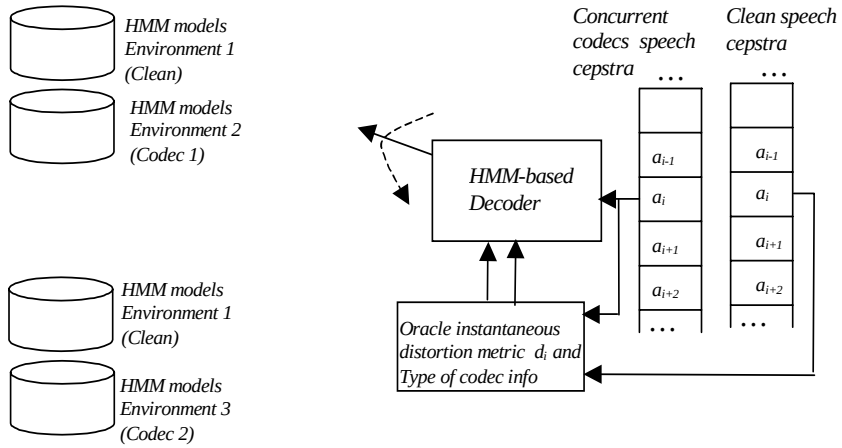

Figure 8.13 Block diagram representing a decoder operating in a Concurrent coding scenario and employing a weighted acoustic modeling technique organized in a structured model fashion, employing oracle cepstral distortion information.

Recognition results using structured acoustic modeling are shown in figure 8.14. We can see that the best performance using the structured acoustic modeling is achieved using 32 Gaussians. In this case, the degradation in performance introduced by concurrent coding was reduced by half (from almost 1.0% to approximately 0.5%). These improvements in the recognition results are comparable to those obtained with the similar GSM oracle experiment, and better than the results obtained in the concurrent coding case when the models are not structured. Thus, it is advantageous to structure the acoustic models when concurrent coding conditions exist, rather than using the multi-style approach to modeling.

Currently, no cellular network standard provides this sort of oracle information. We can summarize from the observations above that in order to have maximum recognition performance a recognition system would need to be provided with cepstral distortion information and information identifying the coding standard employed. The cepstral distortion information would need to be computed either by a cepstral front-end or by a modified coder at the user terminal device.This would add to the computational load of the terminal device. This information, along with the type of codec information would need to be sent through the wireless communication link, thus consuming additional channel bandwidth. The biggest challenge for the practical implementation of this approach is the difficulty involved in the standardization process associated with this type of effort. It would be advantageous for ASR if future standardization efforts of Distributed Speech Recognition front-ends consider the need to include distortion information at the feature level.

8.6.3 Concurrent coding recognition experiments using LTPM information

The previous subsection was focused to establishing a performance upper bound by providing oracle information to the speech recognizer. As we saw in that subsection, the best results were obtained using a system that employs structured weighted acoustic modeling and oracle cepstral distortion information. We now focus on the case

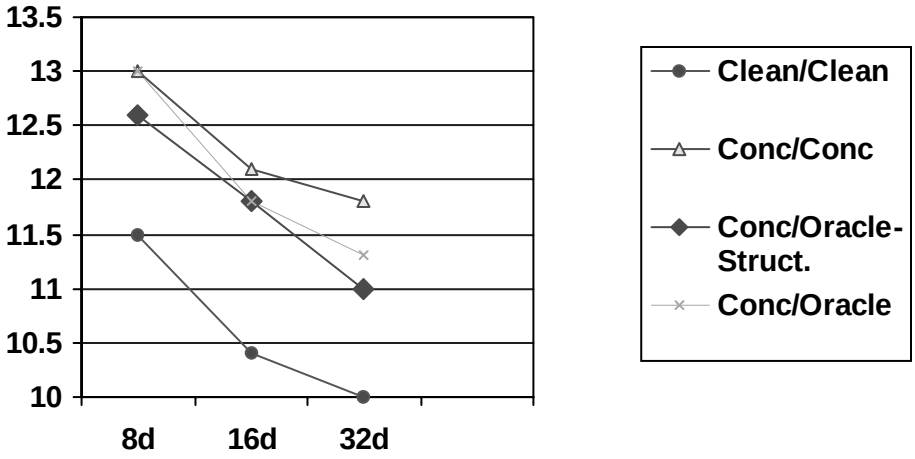

Figure 8.14 Recognition results obtained from experiments implementing a decoder operating in a Concurrent coding scenario and employing a weighted acoustic modeling technique organized in a structured model fashion, employing oracle cepstral distortion information.

where concurrent codecs are affecting the speech signal but no oracle information of any sort is employed during recognition. In order to make the approach codec independent we employ LTPM information as a substitute of cepstral distortion. Because the type of coding is not known to the recognizer, we employ multi-style weighted acoustic modeling instead of structured weighted acoustic modeling. In order to avoid the dependence on the availability of clean data we developed our acoustic models using LTPM information during training, the same way we did in section 8.5.1.

Figure 8.15 shows the results of these experiments using no oracle information. We can see that the best results were obtained using 16x2 Gaussians per mixture, which are comparable to the 32d case because of the retraining pass involved. In this case, our approach reduced the degradation gap by around 0.3% absolute or approximately 15% relative to the size of this gap. These gains are modest when compared with the results obtained using the oracle information, indicating that in a real-implementation system, a further reduction in the error rate can be realized through modifications to front-end protocols that provide the oracle information to the ASR decoder. Similar conclusions can be made based on the experiments performed with only GSM coding (described

previously in Sections 8.2 and 8.5.1) when we compared the results obtained using oracle cepstral distortion information (Section 8.2) with those obtained with LTPM based distortion estimates (Section 8.5.1).

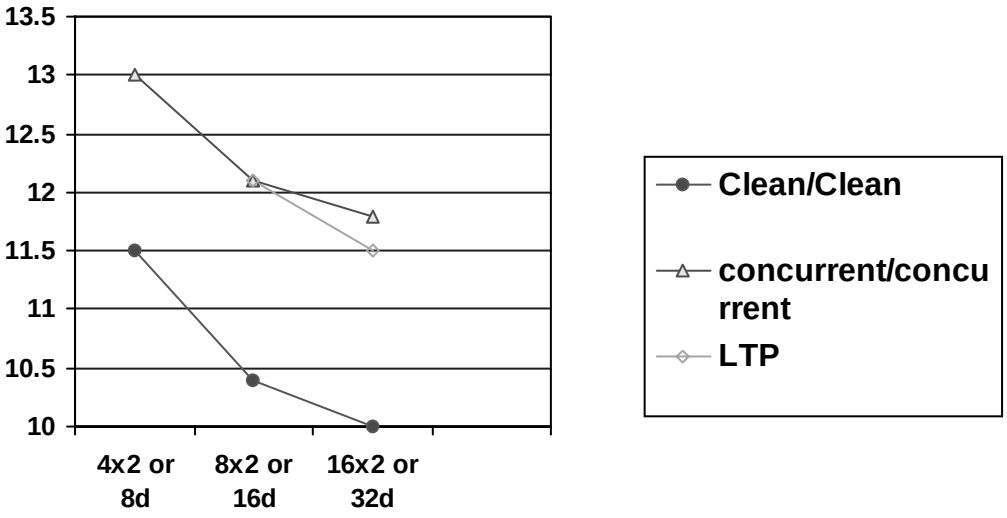

Figure 8.15 Recognition results obtained from experiments implementing a decoder operating in a Concurrent coding scenario and employing a weighted acoustic modeling technique employing instantaneous distortion information derived from long-term predictability analysis.

8.7 Summary

In this chapter we described the method of weighted acoustic modeling based on instantaneous distortion information. In previous chapters we had observed that a significant reduction of the degradation in performance due to GSM coding was achieved by employing average phonetic distortion information and weighted acoustic modeling. We expected to extend the reduction of this degradation by employing instantaneous distortion information instead of average phonetic information. At the beginning of this chapter, in Section 8.2 we established the bounds on error rate reduction using instantaneous distortion by employing cepstral distortion information. The main challenge of this approach was to establish a way to estimate this instantaneous distortion information.

We proposed the use of long-term predictability information (LPTM) and re-coding sensitivity information to estimate the instantaneous distortion introduced by the codec. The main challenge of these approaches was the non-linear relation that exists between these estimates and the distortion introduced to the cepstral vector. When we applied these estimates to recognition the reduction in error rate was considerably smaller compared to the results we had obtained in the oracle experiments. It is clear therefore, that significant improvement of WER can be achieved if this information (i.e., instantaneous cepstral distortion introduced by the codec) was computed at the terminal device and transmitted along the codec information.

Similarly, the results obtained in the context of concurrent coding indicate the advantages of employing cepstral distortion information to the acoustic modeling method. The LTPM based weighted acoustic modeling did not perform as well as the cepstral distortion experiment. In addition, this decoding scenario also benefited from the availability of codec identity information that helped the recognizer structure its acoustic models. Unlike oracle cepstral distortion information, there is no computational cost associated with this information but it consumes channel bandwidth. The utility of this information has been demonstrated in this chapter.

Chapter 9
Summary of results and conclusions

In this thesis we focused on the effect of speech coding algorithms on speech recognition systems. This problem is of fundamental importance to speech recognition in mobile environments as it constitutes the common denominator in all the modalities of mobile ASR. One of the principal factors causing recognition performance to degrade in mobile applications is the codec itself. Additional factors, like environmental noise and carrier interference compound the problem of recognition of reconstructed compressed speech. The focus of speech coding in the context of ASR presented in this thesis should constitute a starting point to further studies of the joint effect of coding and environmental effects in mobile environments. In this chapter, we summarize the findings and contributions of the thesis and present directions for future work.

9.1 Summary of findings and contributions of this thesis

- **Source of the ASR degradation when Speech coding is present**: Both the quantization of the LPC or short-term information and the long-term information affects recognition. However, the quantization of the long-term information has a larger impact on recognition than the quantization of the short-term information. This happens in spite of the fact that a much larger percentage of the bits in a speech codec are allocated to represent the residual signal, and that the residual information carries information of little relative utility for recognition.

- **Assimilation of the cepstral front-end and the Speech Codec:** Because of the larger distortion introduced to the short-term residual signal due to quantization of the long-term information, we can circumvent the effect of this quantization distortion on recognition if we selectively derive a cepstral feature from the codec parameters. This technique completely eliminates the effect of the degradation in

recognition accuracy introduced by coding. This approach has direct implications on the voice over IP and possible future research efforts in distributed speech recognition.

- **Phonetic identity and patterns of LTP quantization-induced distortion:** Based on histograms of the distribution of the distortion introduced by the RPE-block we were able to associate phonetic categories and long-term residual distortion. An approximate relation between phone recognition accuracy and average distortion was observed.

- **Weighted acoustic modeling based on phonetic information:** Based on knowledge of the average amount of distortion introduced by coding to each phone we proposed weighting accordingly the acoustic models representing different distortion environments. This approach reduces the GSM coding performance gap by almost 70%. This technique assumes the availability of clean data. No retraining process is needed, making this technique easy to implement. The weights employed to combine the acoustic models can be obtained directly from histograms of distribution of the distortion in each phone.

- **Weighted acoustic modeling based on instantaneous distortion information:** This method is of relevance when no separate corpus of clean data is available. We proposed two techniques to derive instantaneous distortion estimates: recoding sensitivity and long-term predictability information. In this approach acoustic model retraining is necessary, thus there is a computational overhead *prior* to recognition. The observed gains in recognition accuracy were more limited than using average phonetic information. However, we found that it is advantageous to use this method from a computational viewpoint during recognition.

- **Utility of oracle distortion information during recognition**: We demonstrated that if the front-end or codec at the terminal device provides the speech recognizer with oracle cepstral distortion information along with the distorted or compressed

speech signal, the degradation can be reduced by up to 50%. Along with the results related to the assimilation of the cepstral front-end and the speech codec, these results suggest that future standards can benefit from considering designing both ASR front-ends and speech coding techniques jointly.

- **Weighted acoustic modeling applied to concurrent coding ASR:** For the case of concurrent codecs we applied LTP based distortion estimates obtaining modest results. However, more substantial results are obtained in concurrent coding experiments when oracle information is provided to the recognizer. The best results are obtained when the oracle information provided to the recognizer also reveals the type of codec so that the weighted acoustic modeling is performed using matched codec conditions, in a way that we called structured acoustic modeling.

9.2 Directions for future work

- **Coding standards and protocols for broadband voice and audio, telephony and other applications over IP:** Speech coding standards for commercial cellular telephony have been following a very well-defined trend (*i.e.* CELP family of coding). The current growth potential of voice over IP technologies and Internet based speech and audio applications will emphasize the need to focus on the differences between the nature of this paradigm and cellular telephony. These differences include issues related to the nature of the network, the nature of the terminal devices etc. These differences will make it necessary to extend the set of analysis and techniques proposed in this thesis to new and upcoming coding standards and environments that will work on speech, music and broadband audio.

- **Unification of distributed speech recognition and speech coding front-ends:** Unified coding and front-end standards might possibly consider scalable front-ends in which the Long Term Residual information is included in the transmitted data only during voice communications. For ASR-focused communications, the LTP

information is not needed and thus, could be omitted. Future work on this area can be guided towards integrated, scalable ASR/codec front-ends which are compatible across themselves (for computational and coding advantages) and scalable (for application versatility purposes).

- **Distributed speech recognition:** Future cellular telephony standards will possibly have provisions for automatic speech recognition front-ends embedded in the terminal device. Designing front-ends that perform robustly under all types of situations, range of users and environments under severe bit-rate and computational constraints makes it necessary to focus on efficient parametrization and robust source coding techniques.

- **Speech coding in the presence of environmental acoustic noise:** The potential speaker's exposure to intense environmental conditions presents an additional challenge to speech recognition in mobile environments. The coding process that was designed to represent the speech signal based on a model of the speech production mechanism is now applied to a signal that fails to meet the production model. The coding and quantization distortion introduced to the reconstructed signal is then larger than that typically introduced during the speech-only scenario. Appendix A outlines the effect of noise on coding based on the analysis presented in Chapter 4. Possible future work on the area includes the application of a model of the degradation in the cepstral domain in a similar way as the work done by Moreno [60] and Acero [1] under linear channel distortion.

Appendix A
Full Rate GSM speech coding under additive noise conditions

In this Appendix we extend the analysis of the effect of GSM coding introduced in Chapter 4, to the case of speech in the presence of additive noise. In accordance with the nomenclature of that chapter, let $s[n]$ be the clean speech signal. Let $s'[n]$ be the observed signal which is the result of the sum of the original speech signal $s[n]$ plus the additive noise signal $x[n]$. Then,

$$s'[n] = s[n] + x[n] \tag{A.1}$$

We can express the Equation above in terms of Equation 4.1,

$$s'[n] = \hat{h}[n] * e[n] + x[n] \tag{A.2}$$

Where $\hat{h}[n]$ is the impulse response of the quantized version LPC synthesis filter of $s[n]$, $e[n]$ is the short-term residual of $s[n]$, and $x[n]$ is the environmental noise.

The LPC analysis block will produce a filter $h'[n]$ that will be the result of the short-term analysis of the overall signal (speech plus noise). In order to simplify the analysis let us assume that the spectral estimate of the LPC filter of the noisy signal can be reasonably approximated by the LPC filter of the clean speech signal, *i.e.*,

$$h'[n] \approx h[n] \tag{A.3}$$

Let $\hat{h}'[n]^{-1}$ be the impulse response of the quantized version of the LPC analysis filter of the noisy speech signal. We can extend the approximation above to the quantized versions of the filters $\hat{h}'[n] \approx \hat{h}[n]$.

The short-term residual of the noisy speech signal $e'[n]$ can then be expressed as:

$$e'[n] = s'[n]*(\hat{h}'[n])^{-1} = e[n] + x[n]*\hat{h}[n]^{-1} \quad \text{(A.4)}$$

The expression above, indicates that the short-term residual of the noisy speech signal is equal to the short-term residual of the original clean speech signal plus a filtered version of the additive noise.

The reconstructed short-term residual of the noisy signal can be expressed in terms of the short term-residual of the noisy signal and the quantization noise due to the coding process of the long-term residual of the noisy signal:

$$\hat{e}'[n] = e'[n] + \Upsilon(r'[n]) \quad \text{(A.5)}$$

We can rewrite Equation A.5 in terms of A.4:

$$\hat{e}'[n] = e[n] + x[n]*\hat{h}[n]^{-1} + \Upsilon(r'[n]) \quad \text{(A.6)}$$

The reconstructed long term residual of the noisy speech signal is then shown in Equation A.6. The quantization noise due to the RPE coding process of the long-term residual is a function of the long term-residual or the unpredictable part of $e'[n]$.

Because the signal $s'[n] = s[n] + x[n]$ is expected to be less predictable than the original speech signal $s[n]$ particularly for cases of uncorrelated additive noise, we can expect the following to be true:

$$\Upsilon(r'[n]) > \Upsilon(r[n]) \tag{A.7}$$

On the decoder side, the synthesized speech signal will be equal to the reconstructed residual signal convolved by the impulse response of the quantized LPC synthesis filter:

$$\hat{s}'[n] = \hat{h}[n]*\hat{e}'[n] \tag{A.8}$$

Substituting Equation A.6 into Equation A.8,

$$\begin{aligned} \hat{s}'[n] &= \hat{h}[n]*(e[n] + x[n]*\hat{h}[n]^{-1} + \Upsilon(r'[n])) \\ &= \hat{h}[n]*e[n] + \hat{h}[n]*x[n]*\hat{h}[n]^{-1} + \hat{h}[n]*\Upsilon(r'[n]) \\ &= s[n] + x[n] + \hat{h}[n]*\Upsilon(r'[n]) \end{aligned} \tag{A.9}$$

Equation A.9 means that the quantized version of the speech plus noise signal is equal to the original speech signal, plus the original noise signal, plus the filtered version of the quantization term introduced by the RPE block.

From Equations A.7 and A.9 we can see that coding speech in the presence of noise introduces a quantization noise term larger than the quantization noise introduced by the speech signal itself. This compounds the problem of ASR of coded speech in the presence of additive noise.

Bibliography

[1] Acero A., “Acoustical and Environmental Robustness in Automatic Speech Recognition”, Kluwer Academic Publishers, Boston, 1993.

[2] Atal B.S., “Effectiveness of Linear Prediction Characteristics of the Speech Wave”, J. Acoust. Soc. Am. Vol. 55, No. 6 pp. 1304-1312, June 1974.

[3] Aftelak S., “New Speech Related features in GSM”, in: GSM: Evolution Towards 3rd Generation Systems. Edited by Zvonar Z., Jung P. and Kammerlander K. Kluwer Academic Publishers 1999.

[4] Akaiwa Y., “Introduction to Digital Mobile Communication”. Wiley Series in Telecommunications and Signal Processing. J. Proakis, series editor. John Wiley and Sons 1997.

[5] Bahl L.R., Jelinek F., Mercer R.L., “A maximum likelihood approach to continuous speech recognition”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-5, pp. 179-90, March 1983.

[6] Beyerlein P., “Discriminative Model Combination”, in: Proc. ICASSP ‘98, May 12-15 1998, Seattle, Washington, USA. Vol.1 pp. 481-484.

[7] Bourlard H., Dupont S., Hermansky H., Morgan N., “Towards subband-based speech recognition”, Signal Processing VIII, Theories and Application Proceedings of EUSIPCO-96, Eighth European Signal Processing Conference, p. 3 vol. lxiii+2144, 1579-82 vol.3 Italy.

[8] Bourlard H., Dupont S., “Subband-based speech recognition”, 1997 IEEE International Conference on Acoustics, Speech, and Signal Processing, p., 1251-4 vol.2 Munich, Germany; 21-24 April 1997.

[9] Calhoun G., “Wireless access and the local telephone network”, Boston: Artech House, c1992.

[10] Campbell Jr. J.P., Tremain T.E., Vanoy C. Welch, “The DoD 4.8 KBPS Standard (Proposed Federal Standard 1016)”, in Advances in Speech Coding edited by Atal B. Cuperman V. and Gersho A. Kluwer Academic Publishers 1989.

[11] Cox R.V., “Speech Coding Standards”, in Speech Coding and Synthesis, Elsevier Science B.V., W.B. Kleijn and K.K.Paliwal (editors), Amsterdam 1995.

[12] Das S., Lubensky D., Cheng W., “Towards Robust Speech Recognition in the Telephony Network Environment- Cellular and Landline Conditions”, EUROSPEECH 1999.

[13] Davis S., Mermelstein P., “Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences”, IEEE Transactions on Acoustics Speech and SIgnal Processing, v. ASSP-28, n.4, p.357-366, Aug.1980.

[14] Degener J., Bormann C. GSM speech compression software implementation ftp://ftp.cs.tu-berlin.de/pub/local/kbs/tubmik/gsm/, 1992.

[15] Delphin-Poulat L., Mokbel C., “Frame-Synchronous adaptation of cepstrum by linear Regression”, IEEE Proc. ASRU 1997.

[16] Dempster A.P.; Laird N.M.; Rubin D.B., “Maximum likelihood from incomplete data via the EM algorithm”, J. Roy. Stat. Soc., 39 (1): 1-38, 1977.

[17] De Veth J., Boves L., “Channel normalization techniques for automatic speech recognition over the telephone”, Speech Communication, vol.25, no.1-3, p. 149-64.

[18] Digalakis V., Neumeyer L., Perakakis M., 1998 Quantization of Cepstral Parameters for Speech Recognition over the World Wide Web”, ICASSP 1998.

[19] Duda and Hart, “Pattern Classification and Scene Analysis”, Wiley-Interscience, 1973.

[20] Dufour S., Glorion C., Lockwood P. “Evaluation of the Root-Normalised Front-end (RM_LFCC) for SPeech Recognition in Wireless GSM Network Environments”, ICASSP 1996.

[21] Elvira J.M., Torrecilla, “Name dialing using final user defined vocabularies in mobile (GSM & TACS) and fixed telephone networks”, ICASSP 1998.

[22] Euler S., Zinke J., “The Influence of Speech Coding Algorithms on Automatic Speech Recognition”, IEEE Transactions on Acoustics, Speech and Signal Processing. Vol ASSP- No. 1994.

[23] European Telecommunication Standards Institute, “European digital telecommunications system (Phase 2); Full rate speech processing functions (GSM 06.01)”, ETSI 1994.

[24] Fissore L., Ravera F., Vair C., “Speech Recognition over GSM: SPecific features and Performance Evaluation”, Proceedings of the Workshop on Robust Methods for Speech Recognition in Adverse Conditions, Tampere, Finland 1999.

[25] Furui S., “Cepstral analysis technique for automatic speaker verification”, IEEE ASSP 29:254-272, April 1981.

[26] Gales M.J.F., Young S.J., “Robust continuous speech recognition using parallel model combination” IEEE Transactions on Speech and Audio Processing, Sept. 1996 vol.4, no.5, p. 352-9.

[27] Gallardo-Antolin A., Diaz-de-Maria F. Valverde-Albacete F., “Recognition from GSM Digital Speech”, ICSLP 1998.

[28] Gallardo-Antolin A., Diaz-de-Maria F., Valverde-Albacete F., “Avoiding Distortions due to Speech Coding and Transmission Errors in GSM ASR Tasks”, ICASSP 1999.

[29] Galler M.; Junqua J.-C., “Robustness improvements in continuously spelled names over the telephone”, ICASSP1997.

[30] Gersho A., “Concepts and Paradigms in Speech Coding”, in: Speech Recognition and Coding, New Advances and Trends. Edited by Rubio-Ayuso j.A. and Lopez-Soler J.M., NATO ASI Series, Springer 1995.

[31] Gillick L., Cox S.J., “Some Statistical Issues in the Comparison of Speech Recognition Algorithms”, Proc. ICASSP 1989, Vol. 1, pp. 532-535, Glasgow, England.

[32] Gupta S.K., Soong F., Haimi-Cohen R., “High Accuracy Connected Digit Recognition for Mobile Applications”, ICASSP 1996.

[33] Haavisto P., “Speech Recognition for Mobile Communications”, Proceedings of the Workshop on Robust Methods for Speech Recognition in Adverse Conditions, Tampere, Finland 1999.

[34] Haeb-Umback R., “Robust Speech Recognition for Wireless Networks and Mobile Telephony”, EUROSPEECH 97.

[35] Hanson B.A., Applebaum T.H., Junqua J.C., “Spectral Dynamics for Speech recognition under Adverse Conditions”, in Automatic Speech and Speaker Recognition, Advanced Topics, edited by C.H. Lee, F.K. Soong and K.K. Paliwal, Kluwer Academic Publishers.

[36] Hermansky H., “Perceptual Linear Predictive (PLP) analysis of speech”, J. Acoust. Soc Am. 87 (4), April 1990.

[37] Hermansky H., Wan E., Avendaño C. “Noise Suppression in Cellular Communications”, 2nd IEEE Workshop on Interactive Voice Technology for Telecommunications Applications (IVTTA 94) Sept. 1994 Japan.

[38] Hillerbrand F., “The Status and Development of the GSM Specifications”, in: GSM: Evolution Towards 3rd Generation Systems. Edited by Z. Zvonar, P. Jung and K. Kammerlander. Kluwer Academic Publishers 1999.

[39] Huang, X.D., Hwang M.-Y., Jiang L., Mahajan, M. “Deleted interpolation and density sharing for continuous hidden Markov models”, 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings p. 6 vol. lvii+3588, 885-8 vol. 2.

[40] Hwang M.-Y., “Subphonetic Acoustic Modeling for Speaker-Independent Continuos Speech Recognition”, Ph.D. Thesis, School of Computer Science, Carnegie Mellon University, 1994.

[41] Huerta J.M., Stern R.M., “Speech Recognition from GSM Codec Parameters”, Proc. ICSLP-98, 1998.

[42] Huerta J.M., Stern R.M., “Distortion-class weighted acoustic modeling for Robust Speech Recognition Under GSM RPE-LTP coding”, Proceedings of the Robust Methods for Speech Recognition in Adverse Conditions, Tampere Finland 1999.

[43] Itakura, F., “Line spectrum representations of linear predictive coefficients of speech signals,” J.Acoust. Soc. Am., vol. 57, p S35, Apr. 1975.

[44] Jelinek F., Bahl L.R., Mercer R.L., “Design of a linguistic statistical decoder for the recognition of continuous speech”, IEEE Transactions on Information Theory, vol. IT-21, pp. 250-56, 1975.

[45] Jelinek F., “Statistical Methods for Speech Recognition”, the MIT Press, Cambridge Mass. 1997.

[46] Karray L., Ben Jelloun A., Mokbel C., “Solutions for Robust Speech Recognition over the GSM Cellular Network”, ICASSP 1998.

[47] Kemp D., Sueda R., Tremain T., “An Evaluation of 4800 bps Voice Coders”, Proceedings of the IEEE International conference on Acoustics, SPeech and Signal Processing (ICASSP), 1989, p.200-203.

[48] Kleijn W.B., Paliwal K.K, “An introduction to Speech coding”, in Speech Coding and Synthesis, Elsevier Science B.V., W.B. Kleijn and K.K.Paliwal (editors), Amsterdam 1995.

[49] Kleijn W.B., Paliwal K.K., “Speech Coding and Synthesis”, Elsevier Science B.V., Amsterdam 1995.

[50] Kroon P., Deprettere E. F., Sluyter R. F., “Regular-Pulse Excitation - A Novel Approach to Effective and Efficient Multi-pulse Coding of Speech”, IEEE Trans. on Acoustics, Speech and Signal Processing, 34:1054-1063, October 1986.

[51] Kroon P., Kleijn W.B., “Linear-Prediction base Analysis-by-Synthesis Coding”, in Speech Coding and Synthesis Kleijn-Paliwal editors 1995.

[52] Lawrence C., Rahim M., “Integrated bias Removal techniques for robust speech recognition”, Computer Speech and Language (1999) 13, 283-298.

[53] Leggetter, P.C., Woodland P.C., “Maximum Likelihood Linear Regression for Speaker Adaptation of Continuous Density Hidden Markov Models”, Computer Speech and Language, vol. 9, pp. 171-185.

[54] Lilly B. T., Paliwal K. K., “Effect of Speech Coders on Speech Recognition Performance”, Proc. ICSLP-96, 1996.

[55] Lowerre B., “The Harpy Speech Understanding System”, Ph.D. thesis, Computer Science Department, Carnegie Mellon University, Apr. 1976.

[56] Markel J.D., Gray Jr. A.H., “Linear Prediction of Speech”, Springer-Verlag 1976.

[57] Ming J., Hanna P., Stewart D., Owens M., Smith J. “Improving Speech Recognition Performance by Using Multi-model approaches”, ICASSP 1999.

[58] Mokbel C., Jouvet D., Monne, J., “Deconvolution of telephone line effects for speech recognition”, Speech Communication, vol.19, no.3, p. 185-96.

[59] Mokbel C., Mauuary L., Jouvet, Monne J., Sorin C., Simonin J., Bartkova K., “Towards Improving

ASR Robustness for PSN & GSM Telephone Applications", 2nd IEEE Workshop on Interactive Voice Technology for Telecommunications Applications (IVTTA1994).

[60] Moreno P.J., "Speech Recognition in Noisy Environments", Ph. D. Thesis. Department of Electrical and Computer Engineering. Carnegie Mellon University 1996.

[61] NIST National Institute of Standards and Technology, The TIMIT CDROM, 1989.

[62] NIST, The Resource Management Corpus (RM1) November 1989.

[63] Nokia, "Wireless Data Evolution White Paper: Nokia's vision for wireless data in GSM networks today and tomorrow", April 1998, available for download at http://www.forum.nokia.com/library/library.html

[64] Oliphant M. W., "The mobile phone meets the Internet", IEEE Spectrum, August 1999.

[65] Oppenheim A.V., Schafer R.W. "Discrete Time signal Processing", Prentice Hall.

[66] Paliwal K.K., Kleijn W.B. "Quantization of LPC Parameters", in Speech Coding and Synthesis Edited by W.B. Kleijn and K.K. Paliwal, Elsevier Science B.V. 1995.

[67] Paping M., and Fahnle T. "Automatic detection of Disturbing Robot Voice and Ping Pong Effects in GSM Transmitted Speech", Eurospeech 1997, Greece.

[68] Puel J., André-Obrecht "Cellular Phone Speech Recognition: Noise Compensation vs. Robust Architectures", Eurospeech 1997, Greece.

[69] Rabiner L., Juang, B-H., "Fundamentals of Speech Recognition", Prentice Hall, Englewood Cliffs 1993.

[70] Rabiner L.R., Schafer R.W., "Digital Processing of Speech Signals", Prentice-Hall, Englewood Cliffs New Jersey 1978.

[71] Ramaswamy G., Gopalakrishnan P., "Compression of Acoustic Features for Speech Recognition in Network Environments", ICASSP 98.

[72] Salonidis T., Digalakis V. "Robust Speech Recognition for Multiple Topological Scenarios of the GSM Mobile Phone System", ICASSP 1998.

[73] Soulas T., Mokbel C., Jouvet D., Monné J. "Adapting PSN recognition Models to the GSM Environment by Using Spectral Transformation", ICASSP 1997.

[74] Trancoso I.M., "An Overview of Different Trends on CELP Coding", in: Speech Recognition and Coding, New Advances and Trends. Edited by Rubio-Ayuso J. and Lopez-Soler J.M., NATO ASI Series, Springer 1995.

[75] Tucker R., Robinson T., Christie J., Seymour C., "Compression of Acoustic Features- Are Perceptual Quality and Recognition Performance Incompatible Goals", Eurospeech 1999, Hungary.

[76] Vary P., Hofmann R., Hellwig, "A Regular-Pulse Excited Linear Predictive Codec", Speech Communication 1988.

[77] Viterbi, A.J., "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm", IEEE Transaction s on Information Theory, IT-13, pp. 260-67, 1967.