Towards an Improved Model of Dynamics for Speech Recognition and Synthesis

by

Hongwei Hu

A thesis submitted to
The University of Birmingham
for the degree of
DOCTOR OF PHILOSOPHY

Electronic, Electrical & Computer Engineering
School of Engineering
The University of Birmingham
January 2012

Abstract

This thesis describes the research on the use of non-linear formant trajectories to model speech dynamics under the framework of a multiple-level segmental hidden Markov model (MSHMM). The particular type of intermediate-layer model investigated in this study is based on the 12-dimensional parallel formant synthesiser (PFS) control parameters, which can be directly used to synthesise speech with a formant synthesiser. The non-linear formant trajectories are generated by using the speech parameter generation algorithm proposed by Tokuda and colleagues. The performance of the newly developed non-linear trajectory model of dynamics is tested against the piecewise linear trajectory model in both speech recognition and speech synthesis. In speech synthesis experiments, the 12 PFS control parameters and their time derivatives are used as the feature vectors in the HMM-based text-to-speech system. The human listening test and objective test results show that, despite the low overall quality of the synthetic speech, the non-linear trajectory model of dynamics can significantly improve the intelligibility and naturalness of the synthetic speech. Moreover, the generated non-linear formant trajectories match actual formant trajectories in real human speech fairly well. The N-best list rescoring paradigm is employed for the speech recognition experiments. Both context-independent and context-dependent MSHMMs, based on different formant-to-acoustic mapping schemes, are used to rescore an N-best list. The rescoring results show that the introduction of the non-linear trajectory model of formant dynamics results in statistically significant improvement under certain mapping schemes. In addition, the smoothing in the non-linear formant trajectories has been shown to be able to account for contextual effects such as coarticulation.

Dedication

To my parents.

Acknowledgments

Firstly, I am most grateful to Martin Russell, my PhD supervisor, for his guidance, support and help throughout my time as a student at EECE and this long thesis work. I really appreciate the effort he made to photograph his comments for me when I was in China writing up this thesis. Thanks to Ji Ming, Neil Cooke and Peter Jančovič for their constructive feedback on my research, and Sridhar Pammu for his patience in answering my questions about the cluster, Linux and computing in general. Thanks to my colleagues, Ying, James, Shona, Richard, Xin, Mün and Soud, and Mary at the PG Office, for their help during the past several years.

I wish to thank my parents for their continuous encouragements over the years. I can not imagine how this thesis could possibly have been done without their support. Finally, special thanks go to Xiaoyan, for being such an amazing wife, for what she has done for me, our son and the family.

Declaration

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

The year 1968 saw a Stanley Kubrick epic – *2001: A Space Odyssey*, which stars a super computer called HAL that can speak naturally and respond properly to spoken language. Moreover, this highly intelligent machine is capable of lipreading. This classic sci-fi film, in a sense, vividly depicted how man would communicate with machines in the future. To date, voice, lip shape, gesture, emotion and eye movement (Cooke 2006) have all been studied for human-computer interaction. Speech, as the primary means of communication for people, is a major modality of research in natural multimodal human-machine communication.

1.1 Background

Computer speech recognition and synthesis are two key component technologies in natural human-machine interaction. The former deals with the problem of automatic conversion of an acoustic speech signal into text, while the latter performs the reverse task. Current automatic speech recognition (ASR) systems are firmly based on statistical methods (Jelinek 1998), and the use of hidden Markov models (HMMs) (Rabiner 1989) for acoustic modelling predominates in large-vocabulary continuous speech recognition (LVCSR, see, e.g., Young (1995)). On the other hand, the most successful approach to speech synthesis has been based on unit selection techniques (Hunt & Black 1996), though statistical parametric speech synthesis (Black, Zen & Tokuda 2007), most notably HMM-based synthesis (Tokuda, Yoshimura, Masuko, Kobayashi & Kitamura 2000), has recently become increasingly popular. With the emergence of the trajectory HMM (Tokuda, Zen & Kitamura 2003), speech recognition and synthesis began to converge (Zen, Tokuda & Kitamura 2007, Zhang 2009).

Conventional HMMs, whilst popular in both speech recognition and synthesis, treat speech as a surface phenomenon, and take little account of the underlying speech production mechanism (King, Frankel, Livescu, McDermott, Richmond & Wester 2007). The standard HMM framework assumes that the basic structure of speech consists of a sequence of stationary segments of variable duration, with instantaneous transitions between them. All variations from this underlying structure, either durational or segmental, are treated as noise, rather than as the consequence of modelable articulatory phenomena.

To illustrate this point, consider how classical HMMs account for coarticulation – i.e. the phenomenon whereby the articulation of individual segments is almost always influenced by the articulation of neighbouring segments, often to the point of considerable overlapping of articulatory activities (Clark & Yallop 1990). An example showing the effect of coarticulation is the sound [æ̃][1] in a word like 'cant', where [æ] becomes nasalised when the velum lowers prematurely to get ready for the articulation of the following nasal [n]. Taking the traditional "beads-on-a-string" (Ostendorf 1999) approach in which words are represented as a sequence of phonemes[2], conventional HMMs deal with coarticulation by using context-dependent (CD) phone models, such as biphones, triphones and even higher models. A triphone, for instance, takes the form of [x-y+z] in which the given phoneme [y] occurs immediately after an [x] and before a [z]. Different models would therefore exist for the [y] for every unique pair of left and right neighbours. In hidden Markov modelling of speech, virtually all phonetic and phonological variation is dealt with by using these CD models.

Unfortunately, the increased level of detail of context-sensitivity resulting from the use of CD models is achieved at the cost of a huge explosion in the number of model parameters, and consequently a vast quantity of training data is required to enable robust estimation of all parameters. For large vocabulary recognition which includes a variety of speakers and speaking styles, it is impossible to obtain sufficient training data to cover all possible contexts. A number of methods have therefore been exploited to overcome the need for very

[1] The phonetic notation in the text follows the International Phonetic Alphabet (IPA) system (IPA 1999).

[2] The term *phoneme* denotes the smallest segmental unit of sound employed to form meaningful contrasts between utterances (IPA 1999), whereas the term *phone* refers to a particular acoustic realisation of a phoneme.

large training corpora by “tying” parameters between HMMs or between states. For example, generalised triphones (Lee 1990) can be formed by merging similar CD models, and phonetic decision trees (Young, Odell & Woodland 1994) are usually employed as a means of choosing similar states which then share training data and parameters. Alternatively, triphonic models can be composed from models of less context-dependency (e.g., biphones) based on Bayesian statistics, with the result that the number of free parameters is reduced by higher than an order of magnitude (Ming, O’Boyle, Owens & Smith 1999).

Indeed, the use of CD models has been shown to be of great benefit in improving recognition performance, compared with the use of context-independent (CI) models. However, it is an extravagant approach to the coarticulation problem in acoustic-phonetic modelling (Richards & Bridle 1999) and always suffers from data-sparsity problems unless unlimited training data is available. The fact that coarticulation is an inherent phenomenon in speech production implies that it would be most straightforward modelled in the articulatory domain where it occurs. Again taking the nasalised sound [æ̃] in the word ‘cant’ as an example, the process which transforms [æ] into [æ̃] could be described concisely in terms of the spreading of one distinctive feature (see, e.g., Chomsky & Halle (1968)), i.e. the velic aperture (Browman & Goldstein 1992), from [n] backwards to [æ]. Intuitively, an appropriate model of articulator dynamics should be able to characterise contextual effects more efficiently.

An explicit model of dynamics at the production level is expected to aid recognition, especially recognition of conversational speech, which, in comparison with read speech, contains substantial variations caused by increased coarticulation, highly variable speaking rate and various types of disfluency. All these make automatic recognition of conversational and spontaneous speech a big obstacle to the ultimate goal of computer recognition – i.e., be unconstrained, real-time, with human-like performance. Current ASR systems have difficulties in coping with these less controlled task domains, such as automatic transcription of the Switchboard conversational telephone speech (CTS) (Godfrey, Holliman & McDaniel 1992, LDC 1995). Performance in terms of word error rates (WERs)[3] on the orig-

[3] WER is the primary measure of speech recognition performance, which is defined by WER $= \frac{N_D+N_S+N_I}{N} \times 100\%$, where N_D, N_S and N_I donate the number of word deletion, substitution and insertion errors respectively in the recognition output and N is the total number of words in the reference transcripts.

inal Switchboard corpus were initially high, with a reported WER of 67% (Young, Woodland & Byrne 1994) using the hidden Markov model toolkit (HTK)[4] recogniser that gave state-of-the-art performance on read speech from the North American Business (NAB) news texts (e.g., a WER of 7.2% was achieved in the 1994 NIST benchmark test on the NAB read-sentence corpus using the HTK recogniser (Woodland, Leggetter, Odell, Valtchev & Young 1995)). More recent research has reduced these high error rates for the Switchboard-1 recognition task to the vincinty of 19% by using more training data and employing advanced adaptation and speech modelling techniques (Hain, Woodland, Evermann, Gales, Liu, Moore, Povey & Wang 2005). Nevertheless, there is still a considerable performance gap between computer recognition of read and conversational speech. Further, recognition performance by machines still falls far below that of humans – e.g., subjective listening experiments showed that error rates of humans on data from the NAB read speech corpus and the Switchboard database were typically 1% and 4%, respectively (Lippmann 1997).

Model adaptation techniques are commonly used to accommodate systematic changes caused by differences between speaking styles (e.g., read/spontaneous speech), or inter-speaker differences which result from physiological factors, such as the differences between adult's vocal tract and that of a children. In principle, model adaptation attempts to re-estimate a large number of model parameters using only a small amount of adaptation data from the target domain. This is usually achieved by means of adjusting the model parameters using linear transformations which are calculated to maximise the likelihood of the adaptation data (see, e.g, Leggetter & Woodland (1995)), or maximum *a posteriori* (MAP) estimation (Gauvain & Lee 1994) which intends to maximise the posterior distribution of the model parameters to be estimated for which a prior distribution is assumed.[5] Although these approaches are useful in improving ASR performance, they cannot characterise, explicitly, the underlying production strategies which cause the differences. Furthermore, model adaptation has its own problems that need to be solved. For instance, supervised adaptation requires correctly labelled adaptation utterances, and it is always difficult to collect a large

[4] The HTK (Young, Evermann, Gales, Hain, Kershaw, Moore, Odell, Ollason, Povey, Valtchev & Woodland 2005) is one of the most widely used software tools for speech recognition research today.

[5] Model adaptation is used in the synthesis experiments to mimic an individual's voice (see Chapter 6).

number of such utterances; unsupervised adaptation, on the other hand, usually resorts to recognition hypotheses as the supervision source, which includes not only the correct labels but also the recognition errors (Furui 2009).

Rather than relying entirely on generic statistical modelling techniques to accommodate acoustic variability with ever lager training databases, it should be better to directly model the underlying speech production mechanisms which might have caused the variability. In the terminology of Bourlard, Hermansky & Morgan (1996, paraphrasing Jordan Cohen), this is related to the distinction between *speech description* (which we usually do by increasing the number of models and parameters) and *speech modelling* (which we all would like to do by extracting the underlying properties and invariants of the speech signal). Explicit modelling of speech dynamics in an articulatory-based space offers the chance to characterise many phenomena which arise from simple changes in the production mechanism, yet exhibit intricate variations in the surface description of speech. Acoustic features of speech, for example, which are typically derived from short-term log power spectra, reflect articulatory dynamics indirectly, often as movement across (rather than within) frequency bands (Jackson & Russell 2002), resulting in complex paths in the acoustic feature space.

Despite the knowledge we have about speech production and its potential benefits in improving ASR performance,[6] information at this deeper level is not always available in practice. From the perspective of accessibility and practicability, the acoustics is still the prevailing information source for most ASR systems to work with. Actual articulation data, such as the Electromagnetic Articulograph (EMA) data from the MOCHA corpus (Wrench 2001), is very expensive to collect and hence larger scale corpora of this kind are severely limited. Automatic extraction of articulatory features from the acoustic speech signal, or articulatory inversion (Richmond 2001), is a non-trivial pattern processing task and prone to errors. On top of these, even if the articulatory information is available, the underlying model structure of conventional HMMs does not provide a framework for accommodating speech production knowledge since this knowledge is generally not directly applicable to the surface level representations of speech (Russell 1997). Many of those working in the speech

[6] A survey of the use of speech production knowledge to aid ASR has been given by King et al. (2007).

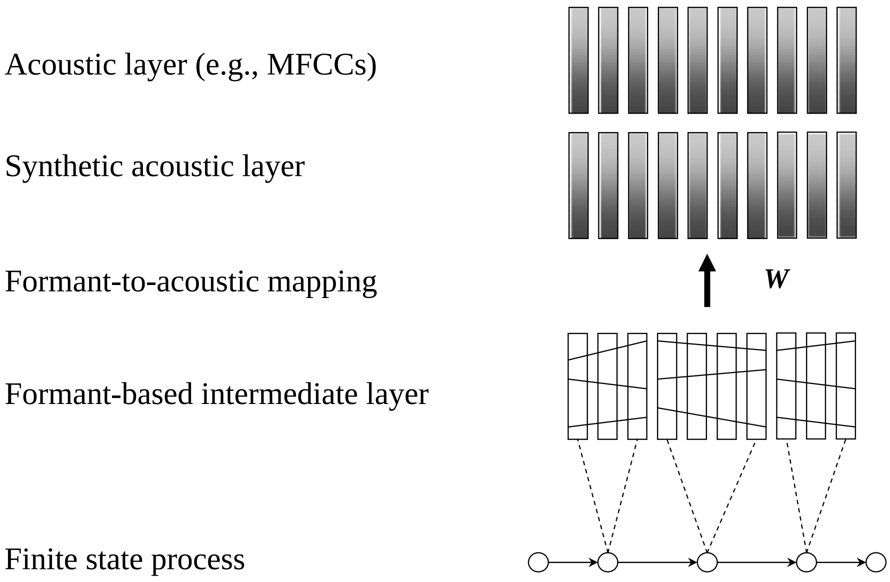

Acoustic layer (e.g., MFCCs)
Synthetic acoustic layer
Formant-to-acoustic mapping
W
Formant-based intermediate layer
Finite state process

Figure 1.1: A linear/linear multiple-level segmental HMM in which the relationship between the symbolic and acoustic representations of a speech signal is regulated by a formant-based intermediate layer.

recognition community believe that new computational paradigms beyond HMMs should be pursued if the ASR problem is to be solved for good, even though these new approaches may lead to the increase in error rates on standard tests in the short term.

In recent years, research at the University of Birmingham, motivated by the considerations mentioned above, has led to a class of multiple-level segmental HMMs (MSHMMs) (Russell & Jackson 2005), in which the relationship between symbolic and acoustic representations of speech is regulated by an intermediate 'articulatory' layer. The incorporation of such an intermediate layer provides the opportunity to model speech dynamics directly in an articulatory-related space. The resulting model is very rich in mathematical structure, which allows many options for the development of alternative articulatory representations, models of speech dynamics and articulatory-to-acoustic mapping schemes.

As a starting point, a simple linear/linear MSHMM was proposed (Russell & Jackson 2005) (see Figure 1.1), in which dynamics were modelled as piecewise *linear* trajectories in the formant-based[7] intermediate layer and projected onto the acoustic layer, e.g., Mel-

[7] Formants are the main resonance modes of the vocal tract, which correspond to the peaks in the short-term spectrum. Formant frequencies are usually referred to as F_1, F_2 and F_3 and so on (in ascending order of the resonance frequencies).

frequency cepstral coeffcients (MFCCs) (Davis & Mermelstein 1980), via a *linear* formant-to-acoustic mapping. The 'synthetic' acoustic layer shown in Figure 1.1 corresponds to the transformed formant trajectories in the acoustic domain (by using a piecewise linear formant-to-acoustic mapping), whereby comparison is made with the actual acoustic observations (i.e. MFCCs). The theory of linear/linear MSHMMs was established, including the derivation of segmental Viterbi decoding and expectation-maximisation (EM) (Dempster, Laird & Rubin 1977) model parameter estimation algorithms (see Chapter 4). A collection of software tools, named 'SEGVit', was implemented to perform MSHMMs training and testing. Extensive phone classification and recognition experiments were conducted and reported (Russell & Jackson 2005, Russell, Zheng & Jackson 2007) on the TIMIT speech corpus (Garofolo, Lamel, Fisher, Fiscus, Pallett, Dahlgren & Zue 1993).

A major motivation for studying these simple linear/linear MSHMMs is to confirm that the introduction of such an intermediate layer does not "hurt" the system performance. An appropriate linear fixed-trajectory segmental HMM (FTSHMM) (Holmes & Russell 1999), which models acoustic features directly using linear trajectories in the acoustic domain, provides a theoretical upper bound on the performance of a linear/linear MSHMM because of the linear nature of the formant trajectories and the linear mappings to the acoustic space.[8] It has been demonstrated that, despite the simplicity of this linear/linear MSHMM system, the phone classification and recognition performance can achieve the upper bound given proper combination of intermediate representations and formant-to-acoustic mappings. In other words, the incorporation of such a formant-based 'articulatory' layer does not compromise the system performance. In fact, compared with a conventional HMM, superior performance can be achieved by using an MSHMM with 25% fewer parameters (Russell & Jackson 2005). These results give us confidence that further improvements in recognition performance may be obtained by using appropriate non-linear trajectory models of dynamics, alternative intemediate 'articulatory' representations or non-linear articulatory-to-acoustic mappings.

The primary goal of this research is to develop an improved, non-linear trajectory model of speech dynamics, relative to a piecewise linear trajectory model as used in previous work

[8] Section 2.3.4 provides a more detailed description of FTSHMMs.

(Russell & Jackson 2005, Russell et al. 2007), and integrate it into the MSHMM framework. Investigating alternative articulatory representations and non-linear articulatory-to-acoustic mappings does *not* form part of the work reported in this thesis. The intermediate 'articulatory' representation concerned in this research continues to be the 12 parallel formant synthesiser (PFS) control parameters[9] from the Holmes-Mattingley-Shearme (HMS) formant synthesiser (Holmes, Mattingly & Shearme 1964), as used in (Russell & Jackson 2005). Although such a formant-based parameterization is not truly articulatory, it will be referred to as an 'articulatory' representation in this thesis to remind us of the motivation for the inclusion of such a layer. A similar intermediate representation has been studied in (Deng 1998, Deng & Ma 2000), which comprises the first three vocal tract resonance (VTR) frequencies – the pole locations of the vocal tract configured to produce speech sounds.

Linear formant-to-acoustic mappings designed for five different phone categorisation schemes, as described in (Russell & Jackson 2005, see Table 4.1 on page 81), will be inherited in this work. For these reasons, the new, non-linear trajectory model will be referred to as a *non-linear*/linear MSHMM in this thesis. However, it should be noted at this point that the relationship between formant and spectrum-based representations of speech is indeed non-linear (Richards & Bridle 1999, Russell et al. 2007), and therefore the use of non-linear formant-to-acoustic mappings in MSHMMs is more compelling. Artificial neural networks (ANNs) have been studied for non-linear mappings in the past. For example, Richards & Bridle (1999) and Deng & Ma (2000) have investigated the use of multilayer perceptrons (MLPs) (see, e.g., Bourlard & Morgan (1994)) to map the formant (or formant-like) space to the acoustic representation of speech. A single MLP was employed by Richards & Bridle (1999), while an ensemble of 10 MLPs, each corresponding to a distinct class of manner of articulation, were used by Deng & Ma (2000). Jackson, Lo & Russell (2002) investigated the use of radial basis function (RBF) networks (see, e.g., Bishop (1995)) for non-linear mapping from formant to short-term spectra, and found that RBF networks consistently mapped better than MLP networks with 10% less the r.m.s error. Up until now, the non-linear mapping component has not yet been integrated into MSHMMs. This research is ongoing.

[9] The 12 PFS control parameters are described in detail in Section 3.2.2 on page 56.

1.2 Non-linear trajectory models of speech dynamics

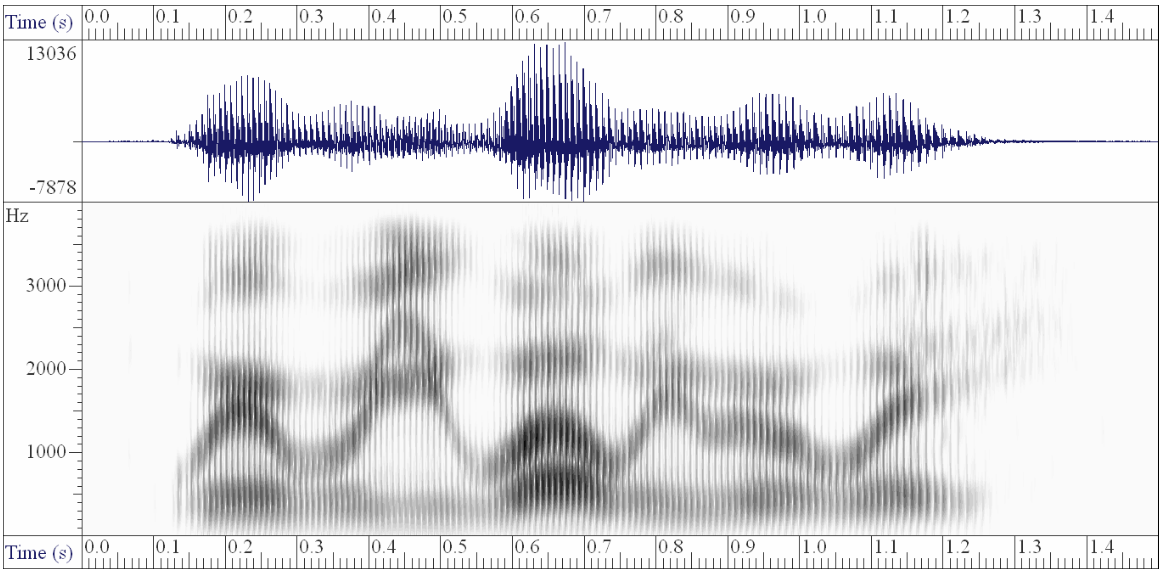


Figure 1.2: Waveform and wide-band speech spectrogram of the sentence (`sx9`) “Where were you while we were away?”, spoken by speaker `msjs1`. This utterance is taken from the TIMIT speech corpus and down-sampled to 8 KHz.

Speech is a continuous signal per se (see Figure 1.2), encoded in which is a sequence of (discrete) symbols conveying some intended message of the speaker. The symbol sequence is ‘transformed’ into the continuous acoustic speech signal through the generally continuous movements of the set of highly constrained components of the vocal tract (tongue, jaw, lips, etc.). The wide-band speech spectrogram provides a convenient representation to inspect speech dynamics due to the fine temporal resolution resulting from a succession of short time windows on the speech signal. As can be seen in the wide-band spectrogram in Figure 1.2, formant trajectories[10] (especially the formant trajectories for the formant frequencies F_1, F_2 and F_3) are mostly smoothly time varying, which reflects the dynamics of the articulators since formant dynamics are closely correlated with the dynamics of the principle articulators (Deng, Acero & Bazzi 2006). For example, the lower F_1 is, the closer the tongue is to the roof of the mouth. Try to pronounce the vowel [i] in a word like ‘beet’ and the sound [ɔ] in a word like ‘bought’, for which the F_1 values are about 300 Hz and 950 Hz, respectively.

[10] A formant trajectory is a succession of values of an individual formant frequency as a function of time, which appears as a dark band in the spectrogram.

One of the limitations of a linear/linear MSHMM presented in (Russell & Jackson 2005, Russell et al. 2007) is the use of piecewise linear formant trajectories to model speech dynamics. Although a piecewise linear trajectory model provides a reasonable 'passive' approximation to the formant trajectories within individual segments, it does not capture the active dynamics of the articulatory system. In particular, no continuity constraints are applied across the segment boundaries. Therefore, a piecewise linear trajectory model is incapable of modelling inter-segmental and long-span dependencies that persist in real human speech.

Of course, there are many good reasons to choose linear models for speech pattern modelling. They are in general simpler and easier to deal with than non-linear models. Digalakis (1992) justified the use of linear models for characterising intra-segmental dependencies presented in the Mel-cepstral parameterization of speech, but found that linear models did not work well for modelling inter-segmental dependencies. Gish & Ng (1993) and Holmes & Russell (1997) also demonstrated the adequacy of a linear trajectory to capture the time-evolving characteristics for most sounds in MFCCs. However, one of the underlying motivations for trajectory-based models is that it should be possible to model coarticulation effects more effectively, instead of having to resort to, for example, triphones, which require large amounts of training data. For this to happen, the trajectories must be able to model realistic transitions between speech units. This is more likely to be achieved by using a more sophisticated, continuous and non-linear trajectory model. The use of non-linear trajectory models will almost certainly incur substantially increased computational time or/and space complexity. Nonetheless, with more computing power at lower price nowadays, this additional cost becomes of less an issue, and thus it should not act as discouragement from exploiting this theoretically more advantageous approach.

In 1995, Tokuda, Kobayashi & Imai (1995) proposed a speech parameter generation (SPG) algorithm to generate a sequence of speech parameters from HMMs which include both static and dynamic features[11]. The speech parameter sequence is generated in such a way that its output probability given the standard HMMs is maximised under the constraints between the static and dynamic features. This algorithm forms the basis of HMM-based

[11] Dynamic features, or delta features, see, e.g., Furui (1986), are discussed in more detail in Section 2.2.3

speech synthesis (Yoshimura, Tokuda, Masuko, Kobayashi & Kitamura 1999). Building on this work, Tokuda et al. (2003) translated the standard HMM into a new trajectory model, referred to as the trajectory HMM. Under such a model, the mean for the spectral parameter vector sequence is most consistent with the static and dynamic constraints (instead of being piecewise constant as in HMMs), which is exactly the same as the speech parameter sequence produced by the SPG algorithm. Tokuda's work is particularly attractive to this research since the generated trajectories are smooth and non-linear, which vary within a state occupancy and are also affected by neighbouring states and models. This is the type of trajectory sought after to characterise the formant dynamics in this work.

This study investigates the use of smooth, non-linear formant trajectories generated by Tokuda's SPG algorithm to characterise formant dynamics. These non-linear formant trajectories are used in place of the piecewise linear trajectories in the intermediate layer of a linear/linear MSHMM. It is hoped that the use of these non-linear formant trajectories, whereby a potentially more advantageous model of formant dynamics is formed, will give rise to further improvement in speech recognition performance, relative to linear/linear MSHMMs based on piecewise linear formant trajectories.

1.3 Implications for speech synthesis

Speech recognition and synthesis, though obviously different in many respects, are in fact both concerned with verbatim translation between symbolic and acoustic representations of speech. Building a unified model for speech recognition and synthesis therefore offers a deeper knowledge sharing between the recognition and synthesis components, and a computational model closer to the human speech skill acquisition process that involves the "simultaneous" learning of speech perception and of speech production (Fallside 1992). Techniques developed for speech recognition, for example speaker adaptation, can be readily used for HMM-based synthesis, generating synthetic speech with various voice characteristics, emotional expressions and speaking styles (Yamagishi, Kobayashi, Nakano, Ogata & Isogai 2009), all of which can be automatically 'learned' from data of the target speaker.

The particular type of intermediate model (i.e., the MSHMM) concerned in this research is potentially suitable for both speech recognition and synthesis. The incorporation of the intermediate, articulatory-related space in an MSHMM explicitly exploits the relationship between symbolic and acoustic representations of speech at a production level. One of the key findings in (Russell & Jackson 2005, Russell et al. 2007) is that high-accuracy phone classification and recognition performance can be achieved when the intermediate layer of an MSHMM is based on all 12 PFS control parameters. In this case, model parameter estimation is equivalent to maximum likelihood (ML) estimation of target and slope values for each PFS control parameter for each state in a set of (context-sensitive) phone-level models. In principle, these 12 PFS control parameters are sufficient to create a 'talker table' to define a 'voice' for a formant synthesiser like (Holmes et al. 1964). Moreover, if speaker adaptation is used in the intermediate layer of an MSHMM to adapt the model to an individual's voice, the resultant synthetic speech should sound like that speaker.

Research elsewhere (Holmes 1973) showed that a parallel formant synthesiser can generate high quality synthetic speech which is almost indistinguishable from the natural in spectrum, waveform, and by earphone listening, provided that the synthetic glottal pulse is derived by inverse filtering a typical natural vowel from the same talker.[12] This is an important result, not only because it demonstrates how good a parallel formant synthesiser can be, but also because it stresses the central importance of getting the synthesiser control parameters (in particular the excitation parameters) right in formant synthesis. From this point of view, the effort of this research is guided towards generating appropriate formant synthesiser control parameters (preferably if this can be done automatically) for the synthesiser, rather than improving the formant synthesiser itself. For these reasons, the ultimate goal for an MSHMM is to build a 'unified' model which can support both high accuracy speech recognition and high quality, trainable and adaptable speech synthesis.

In a linear/linear MSHMM, speech dynamics are modelled as piecewise linear trajectories in the formant-based intermediate layer, without continuity constraints between neigh-

[12] It has to be noted that the experiments reported in (Holmes 1973) were small scale which were based on a very close copy synthesis of real speech samples with synthesiser control parameters, in particular the excitation parameters, carefully tuned by humans.

bouring segments. These assumptions might hold under severely limited situations, though are clearly unrealistic when continuous speech is concerned. From the perspective of a unified speech model, this work is motivated by the belief that a non-linear trajectory model of speech dynamics, relative to a piecewise linear trajectory model, would benefit both speech recognition and synthesis.

Both model based synthesis, for example, HMM-based speech synthesis (Tokuda et al. 2000), and the use of formant information for HMM-based speech synthesis (see, e.g., Acero (1999))[13] have been studied in the past. In either case, Tokuda's SPG algorithm is applied to produce smoothed cepstral or formant trajectories. The novelty of this work is that the SPG algorithm is applied to the 12 PFS data and precedes formant synthesis, and that the synthesis performance is evaluated as another way of assessing the potentially 'unified', non-linear trajectory model.

1.4 Research questions

In the context of an MSHMM, this thesis attempts to address the following questions:

- Can the SPG algorithm (Tokuda, Kobayashi & Imai 1995) be used to generate non-linear formant trajectories suitable for MSHMMs?
- Does the use of non-linear formant trajectories in an MSHMM result in improved speech recognition performance compared with an MSHMM using piecewise linear formant trajectories?
- Is the use of non-linear formant trajectories in an MSHMM able to account for contextual effects, such as coarticulation? If so, is it a more efficient means to coarticulation than the use of traditional context-dependent models?
- Can MSHMMs be used to produce intelligible, good quality and adaptable synthetic speech?

[13] In (Acero 1999), the first three formant frequencies and their corresponding bandwidths, together with the dynamic features, formed the feature vectors in HMMs, and source modelling was dealt with separately.

1.5 Outline of the thesis

The rest of the thesis is organised as follows. Chapter 2 presents a literature review on a variety of models of speech dynamics which are relevant to this work. Chapter 3 describes speech data used in this thesis work in detail, including both acoustic data (MFCCs) and formant data (12 PFS control parameters). The HMS formant synthesiser, driven by the 12 PFS control parameters, is also covered in this chapter as it will be used in synthesis experiments described later in Chapter 6. Chapter 4 provides a detailed description of the MSHMM framework, including the model formalism, training and decoding algorithms. Chapter 5 first describes the SPG algorithm proposed by Tokuda et al., which is adopted to generate the non-linear formant trajectories in this research. The trajectory HMM method (Tokuda et al. 2003) is then discussed, followed by a review of the HMM-based speech synthesis, which is one of the most successful applications of the SPG algorithm. Chapter 6 documents the experimental work and results on assessing the non-linear trajectory model from the perspective of speech synthesis. This chapter gives full details of the speech synthesis system developed for this research and presents subjective and objective experiment results. Chapter 7 concentrates on the application of the non-linear trajectory model to speech recognition. This chapter reports speech recognition experiments and results using non-linear/linear MSHMMs based on the N-best list rescoring paradigm. Chapter 8 concludes this thesis and suggests possible directions for future work.

Chapter 2

Literature review

2.1 Introduction

Modelling speech dynamics is a topic of active research and has been addressed by a number of researchers in a variety of ways. A recent overview of dynamic speech models was given by Deng (2006). A survey of the literature relevant to this thesis work can be divided into the following three parts: 1) conventional HMMs, 2) segmental HMMs (SHMMs), which extend conventional HMMs by modelling sequences of acoustic observation vectors rather than individual feature vectors, and 3) hidden dynamic models (HDMs), in which speech dynamics are modelled in an intermediate, usually articulatory-related, *hidden* space. Standard HMMs will be reviewed first (Section 2.2), as these models lay the theoretical foundation for SHMMs described later in Section 2.3. HDMs, sometimes referred to as *super-segmental* models (Deng, Yu & Acero 2006), are discussed in Section 2.4.

2.2 Hidden Markov models

Although conventional HMMs do not account for speech dynamics adequately, a brief review of standard HMMs and their modelling assumptions (Section 2.2.1) helps to understand where the problem of modelling speech dynamics comes from. After all, it is these models that provide the best compromise between accuracy and efficiency for acoustic-phonetic modelling in current generation LVCSR systems. Hidden semi-Markov models (HSMMs) will be re-visited next (Section 2.2.2), as they will prove useful in describing the SHMM framework. Finally, Section 2.2.3 discusses several issues resulting from the use of dynamic features in HMMs to take account of temporal feature dynamics.

An HMM has a number of states $\mathcal{S} = \{s_1, \ldots, s_N\}$, each representing a distinct 'sound' within an utterance. One or more states are commonly used to characterise a speech segment corresponding to a phone, and then word or sentence models can be constructed by joining the appropriate set of phone models. State transitions are instantaneous and governed by a state transition probability matrix $\boldsymbol{A} = [a_{ij}]_{1 \leq i,j \leq N}$,[1] whereby the durational variability in real human speech is accommodated. However, the resulting state durational model in HMMs is not ideal, as will be discussed later.

Each HMM state is associated with a *frame-level* output probability distribution, whose role is to model variations in the acoustic realisation of that state. Gaussian mixture distributions are most commonly used. In principal, any probability density function (PDF) can be approximated arbitrarily closely by a Gaussian mixture model (GMM) with a sufficiently large number of mixture component. For example, a K-component Gaussian mixture density $b_i(\boldsymbol{o}_t)$ is just a linear combination of K Gaussian PDFs

$$b_i(\boldsymbol{o}_t) = \sum_{k=1}^{K} w_{ik} \mathcal{N}(\boldsymbol{o}_t | \boldsymbol{\mu}_{ik}, \boldsymbol{\Sigma}_{ik}), \tag{2.1}$$

where w_{ik} $(0 \leq w_{ik} \leq 1)$ is the weight of the k^{th} mixture component in state i, $\sum\limits_{k=1}^{K} w_{ik} = 1$, and $\mathcal{N}(\boldsymbol{o}_t | \boldsymbol{\mu}_{ik}, \boldsymbol{\Sigma}_{ik})$ denotes a single multivariate Gaussian density with mean $\boldsymbol{\mu}_{ik}$ and covariance matrix $\boldsymbol{\Sigma}_{ik}$ (which is typically assumed to be diagonal to simplify parameterization and computation), evaluated at $\boldsymbol{o}_t$. As in most cases of mathematical modelling, in the case of acoustic modelling we are interested in characterising the underlying systematic properties of the signal, rather than capturing the specific details (e.g., noise) represented in the training data. Although a single Gaussian PDF may not be sufficiently flexible to accurately account for the actual distribution of the data for a state, increasing mixture components

[1] Let $x_t \in \mathcal{S}$ be the state at time t, then the state transition probability $a_{ij} = P(x_{t+1} = s_j | x_t = s_i)$[2], $0 \leq a_{ij} \leq 1$, and $\forall i = 1, \ldots, N$: $\sum_{j=1}^{N} a_{ij} = 1$.

[2] In this thesis, $P(\cdot)$ and $p(\cdot)$ are used to denote the probability mass and density respectively.

(and hence the model complexity) may lead up to a model which has too much flexibility to generalise well to unseen data. From this perspective, more mixture components do not necessarily guarantee a more accurate model for the problem at hand. On a practical level, the number of free parameters in a mixture model is proportional to the number of mixture components, and therefore the use of mixture models may worsen the data-sparsity problem.

An HMM can be thought of as a generator to give an observation sequence. At time $t = 1$ the model is in state i, where i is determined randomly according to an initial state probability vector $\boldsymbol{\pi}$[3]. A speech vector $\boldsymbol{o}_t$ is then generated randomly according to the state output PDF b_i for this state. At time $t + 1$, the model transits to state j, where j is chosen randomly according to the transition probability matrix $\boldsymbol{A}$. Then a speech vector $\boldsymbol{o}_{t+1}$ is generated randomly and independently of $\{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_t\}$ according to the output PDF b_j for state j. The process goes on and a sequence of observations is produced. This generative view is very important for HMM-based synthesis, as described in Section 5.4.

2.2.1 HMM assumptions

In HMMs, the following assumptions are made: (Holmes & Russell 1999)

- An acoustic observation sequence is produced by a *piecewise stationary* process, for which each stationary region is individually represented by an HMM state, with instantaneous transitions between stationary states.
- Consecutive frames of acoustic observations are statistically independent and identically distributed (IID) given a particular state (a.k.a. *the independence assumption*).
- The underlying (hidden) state sequence is a first order Markov chain, which implicitly gives a *geometric* state duration model.

These assumptions are made for mathematical tractability and computational usefulness, but are inappropriate for modelling speech dynamics. Firstly, human speech production is certainly not a piecewise stationary process, but a continuous and complex one, in which the

[3] $\boldsymbol{\pi} = [\pi_1, \ldots, \pi_N]$, where $\pi_i = P(x_1 = s_i)$, $\forall i = 1, \ldots, N$: $0 \leq \pi_i \leq 1$, and $\sum_{i=1}^{N} \pi_i = 1$.

articulators move slowly and continuously along highly constrained trajectories, each one capable of a limited set of gestures which are organized in an overlapping, asynchronous fashion (Frankel & King 2007). Therefore, speech is, in general, a continuous changing signal, as is shown in Figure 1.2 on page 9.

Secondly, the independence assumption is a major drawback of HMMs as proper models of speech dynamics. The constraints of the human vocal apparatus and more immediately, the restrictions on feature extraction (in which acoustic features are derived from *overlapping* short windows on the acoustic speech signal) ensure that successive acoustic features, especially those within the same phonetic segment, are highly correlated and hence not independent under a Gaussian assumption. This has been justified by Digalakis (1992) on MFCCs and later by Frankel (2003) on the perceptual linear prediction (PLP) (Hermansky 1990) parameters and the EMA data (direct measure of human articulation). Both authors validated the suitability of using the linear regression method to account for the variance of a particular observation within a phone segment and therefore subscribed to linear models on an intra-phone basis in their respective work. By assuming no dependency between observations (other than through the state sequence), HMMs are incapable of modelling the temporal dynamics and time correlation of speech frames that we believe are important in describing speech patterns, not to mention long-term correlations which may spread their effects across the whole utterance, such as speaker characteristics. Moreover, problems associated with the independence assumption are compounded by the use of GMMs in a standard HMM, since the model is free to switch mixture components at every frame, providing a means of modelling acoustic trajectories without knowledge of the underlying process (Iyer, Gish, Siu, Zavaliagkos & Matsoukas 1998). Holmes & Russell (1999) made the case that in a typical speaker-independent HMM system where Gaussian mixture densities are used with different modes corresponding to different speaker sub-populations, the model in effect treats each frame of an utterance as if it may have been spoken by a different speaker.

Finally, the state duration model in an HMM is determined by state self-transition probabilities, with the result that the maximum probability is assigned to a duration of one frame and exponentially decreasing probabilities to longer durations. This corresponds to a geometric distribution for which the probability of the underlying Markov process staying in state i for l ($l = 1, 2, \ldots$) consecutive time units is given by $P(l) = a_{ii}^{(l-1)}(1 - a_{ii})$, where a_{ii} is the self-transition probability for state i. Clearly this geometric distribution is unrealistic for modelling speech patterns since human speech sounds have a typical duration, with shorter and longer durations being less likely.

2.2.2 Hidden semi-Markov models

The weak duration modelling problem can be reduced by adopting an explicit state duration model for each HMM state in place of the state self-transition probability. The resulting model is sometimes referred to as an HSMM (Russell & Moore 1985) or a continuously variable duration HMM (CVDHMM) (Levinson 1986). The behavior of an HSMM as a generative model is different from that of a standard HMM due to the inclusion of explicit duration distributions: at some integer time t, the underlying stochastic process enters state i and then stays in this state for a period of l, which is determined randomly according to the state duration distribution d_i. During this period, a sequence of l observations is generated *independently* according to the state output PDF b_i. At the end of the period the underlying process transits to another state which is chosen randomly according to the state transition probability matrix.

Both non-parametric and parametric duration distributions have been investigated in the past. Non-parametric duration models, such as those proposed in (Ferguson 1980, Ostendorf, Kannan, Kimball & Rohlicek 1992), simply use smoothed relative frequencies. Examples of parametric duration models include the Poisson distribution (Russell & Moore 1985), the Gamma distribution (Levinson 1986) and the Gaussian distribution (Oura, Zen, Nankaku, Lee & Tokuda 2006). Most authors reported positive experimental results when an appropriate explicit duration model, either parametric or non-parametric, was incorporated, though parametric models have less parameters than their non-parametric counterparts.

A problem in explicitly duration modelling is the increased computational complexity in both training and decoding. For example, the time complexity of an HSMM is increased by a factor of L, relative to a standard HMM, in Viterbi decoding, where L is the maximum allowed duration of any state (Mitchell, Harper & Jamieson 1995, Datta, Hu & Ray 2008). A number of modified recursions have been developed to improve the complexity (see, e.g., Yu & Kobayashi (2003), Datta et al. (2008)). Alternatively, Johnson (2005) showed that a standard HMM with a small increase in the number of states is able to closely model actual phone duration distributions on TIMIT and match the performance of a corresponding Gamma duration model. Johnson's approach has the practical advantage that only minimum modification is required to many standard, off-the-shelf software tools for HMMs.

Although HSMMs provide an improved model of state duration, the impact of the independence assumption is still there. Dynamic features are often employed to mitigate the effects of the independence assumption in HMMs, as discussed in the next section.

2.2.3 HMMs with dynamic features

A simple approach to capture inter-frame time dependence is to add first and second order time derivatives (so-called Δ and Δ^2 features, or dynamic features) to the basic static acoustic features (see, e.g., Furui (1986)). In HTK (Young et al. 2005), for example, these dynamic features are calculated by fitting a linear regression over a window covering the Θ preceding and Θ following vectors

$$\Delta \boldsymbol{c}_t = \frac{\sum_{\theta=1}^{\Theta} \theta(\boldsymbol{c}_{t+\theta} - \boldsymbol{c}_{t-\theta})}{2\sum_{\theta=1}^{\Theta} \theta^2}, \tag{2.2}$$

where $\boldsymbol{c}_t$ is the static feature vector at time t and $\Delta \boldsymbol{c}_t$ is a delta coefficient vector computed in terms of the corresponding static features $\{\boldsymbol{c}_{t-\Theta}, \ldots, \boldsymbol{c}_{t+\Theta}\}$. The Δ^2 coefficients can be calculated using the same equation applied to the Δ coefficients. In (Young 1995), for instance, Θ is set to 2, so that the Δ (velocity) and Δ^2 (acceleration) coefficients of each 'augmented' feature vector contain information from the surrounding 4 and 8 frames respectively.

The use of dynamic features proves to be of practical benefit in improving performance for a wide range of applications in speech processing, such as speech recognition (Wilpon, Lee & Rabiner 1991, Lee & Giachin 1991), speaker recognition (Soong & Rosenberg 1988), not to mention HMM-based speech synthesis (Tokuda, Kobayashi & Imai 1995, Tokuda et al. 2000), which is deeply rooted in the application of dynamic features. However, the inclusion of these dynamic features in augmented feature vectors makes both the independence assumption and the piecewise stationarity assumption of HMMs even less appropriate. Not only is each frame of data now 'explicitly' forced to contribute to several neighbouring frames (which further invalidates the independence assumption), but also the resulting model is inconsistent in its assumption that the static, Δ and Δ^2 parameters are constant and possibly non-zero throughout a state occupancy (Russell et al. 2007).

Tokuda et al. (2003) clearly understood the inconsistency in HMMs caused by the use of dynamic features, and reformulated the standard HMM as a new trajectory HMM. The mean trajectory (defined in the static feature space) is calculated in such a way as to explicitly take account of the constraints between static and dynamic parameters (see Section 5.3 for more details on the trajectory HMM). A similar interpretation is given by Bridle (2004), who shows that an HMM which includes dynamic features can be thought of as an 'unconventionally' generative model, from which the generated samples (sequences of static feature vectors) are constrained by the delta distributions, which are independent of statics.

Appending the Δ and Δ^2 parameters to static acoustic features has become a standard exercise in front-end signal processing for ASR. Despite the recent findings on how one can remedy the inconsistency caused by dynamic features in terms of an autoregressive model (Williams 2005), expanding observation space with feature derivatives can be merely counted as another example of speech description (by increasing the number of parameters), rather than speech modelling. It is believed by many researchers that emphasis should be put on speech modelling to make the model more appropriate for speech signals, rather than trying to modify the data to fit the model (Holmes & Russell 1999). SHMMs, as discussed next, are examples which attempt to *model* temporal acoustic feature dynamics.

2.3 Segmental hidden Markov models

2.3.1 General segmental modelling framework

SHMMs are generalised versions of HMMs, which aim to provide an improved model of feature dynamics relative to conventional HMMs, and meanwhile retain advantages of the HMM framework, such as straightforward training and recognition algorithms. In SHMMs, sequences of acoustic vectors, or 'segments'[4], are modelled as homogeneous units so that dependencies between vectors within a segment can be modelled explicitly. Many different types of SHMMs have been studied in the past, and a comprehensive overview has been presented in an earlier review paper (Ostendorf, Digalakis & Kimball 1996).

In general, a state s_i in an SHMM is associated with a state output distribution b_i defined on the set of *sequences* of observation vectors $\left\{\boldsymbol{o}_1^l = \{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_l\}; l \in \mathcal{L}\right\}$[5], where $\mathcal{L}$ is the set of all possible segment lengths (in frames). This differs from an HMM or HSMM, in which state output densities are at the frame level. In addition, an explicit duration distribution d_i is also associated with state i that gives the probability of state duration $l \in \mathcal{L}$, as in the case of the HSMM. Under an SHMM, a sequence of l acoustic vectors $\boldsymbol{o}_1^l = \{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_l\}$ is generated given state s_i according to the density

$$p(\boldsymbol{o}_1^l, l|s_i) = p(\boldsymbol{o}_1^l|l, s_i)p(l|s_i) = b_i(\boldsymbol{o}_1^l)d_i(l). \tag{2.3}$$

An HSMM discussed before (see Section 2.2.2) can be seen as a limiting case of an SHMM for which successive observations are assumed to be IID within given segment boundaries. In this case, Equation (2.3) can be written as

$$p(\boldsymbol{o}_1^l, l|s_i) = b_i(\boldsymbol{o}_1^l)d_i(l) = d_i(l)\prod_{t=1}^{l} p(\boldsymbol{o}_t|s_i). \tag{2.4}$$

Furthermore, if the explicit state duration distribution d_i is assumed to be geometric, then the

[4] In the context of segmental modelling for ASR, the term 'segment' refers to any sequence of frames representing some linguistically meaningful speech unit such as the phone or the subcomponent of the phone.

[5] The sub and superscript are used in the notation $\boldsymbol{o}_1^l$ to explicitly show the start and end time of an individual speech segment $\{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_l\}$.

above segmental model (i.e., the HSMM) reduces to a conventional HMM.

As a generative model, the generation mechanism of an SHMM is described as follows. At some integer time t, the underlying stochastic process enters a state s_i. The stochastic process then stays in this state for a period of l, which is determined randomly according to the state duration distribution d_i. During this period, a sequence of l observations is generated as a *single* sample according to the segmental output distribution b_i. At the end of the period the underlying process transits to a new state according to the state transition probability matrix.

Comparing generation mechanisms of a standard HMM, HSMM and SHMM, it can be noted that one frame of observations is generated by visiting each HMM state, while a variable-length (determined by the explicit duration model) sequence of observations are generated by visiting each state in either an HSMM or SHMM. What is more, in both HMMs and HSMMs, observations are generated independently according to the frame-level state output model, while an observation sequence is generated as a single sample according to the segment-level output densities in SHMMs.

Trajectory-based segmental HMMs

Trajectory-based SHMMs are a broad class of segment models in which acoustic feature dynamics within a segment is modelled by some form of trajectory through the acoustic feature space. The trajectory may be non-parametric or parametric, and in the case of parametric trajectory, the trajectory parameters may be probabilistic or fixed. Typical parametric trajectories studied in the past include constant, linear and higher order polynomials. Observations are assumed to be independent given the segment length, though conditioned on the trajectory of the segment to which they belong. The resulting trajectory model therefore provides a strong continuity constraint between observations in the segment without requiring a complex model of correlations. Parametric SHMMs are discussed first, and these models are reviewed in Sections 2.3.2 - 2.3.5. Non-parametric SHMMs are reviewed in Section 2.3.6 and compared with parametric models in Section 2.3.7.

2.3.2 Static segmental HMMs

Russell (1993) introduced a static SHMM, which adopted two separate components to model *extra-* and *intra-segmental* sources of variability separately. In such a model, a state s_i is associated with a state target PDF b_i (assumed to be a single Gaussian PDF $b_i = \mathcal{N}(\boldsymbol{\nu}_i, \boldsymbol{\eta}_i)$ with mean $\boldsymbol{\nu}_i$ and variance $\boldsymbol{\eta}_i$), which aims to characterise extra-segmental factors of speech that remain fixed over the entire segment, such as speaker identity (when the models are used to characterise speech from a number of speakers) or chosen pronunciation of a speech sound. On arrival at state s_i a target vector $\boldsymbol{c}$ (whose dimension is that of the acoustic feature space) is chosen randomly according to this PDF and then remains static throughout the state occupancy, thus giving a constant trajectory.

The state duration l (hence the trajectory length) is determined randomly according to the duration distribution d_i associated with state s_i. During this period, at each time t $(1 \leq t \leq l)$ the observation $\boldsymbol{o}_t$ is generated randomly and independently of $\{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_{t-1}\}$ according to another Gaussian PDF $\mathcal{N}(\boldsymbol{c}, \boldsymbol{\Sigma}_i)$ with mean $\boldsymbol{c}$ (the target) and fixed variance $\boldsymbol{\Sigma}_i$. This Gaussian distribution is referred to as the intra-segment distribution, which models intra-segmental variations once all sources of extra-segmental variability have been fixed. Figure 2.1 illustrates how the SHMM separates the distribution over the observations into extra-segmental and intra-segmental variations. In contrast, conventional HMMs model these two types of variability together using the mixture components of the output density, which, compounded by the independence assumption, allows extra-segmental factors such as speaker identities to change in synchrony with the frame rate of the acoustic patterns.

Under a static Gaussian segmental HMM (GSHMM, in which both extra- and intra-segmental distributions are assumed to be Gaussian), the joint probability of a segment $\boldsymbol{o}_1^l = \{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_l\}$ of length l and a particular target $\boldsymbol{c}$ given state s_i is

$$p(\boldsymbol{o}_1^l, \boldsymbol{c}|s_i) = d_i(l)b_i(\boldsymbol{c})p(\boldsymbol{o}_1^l|\boldsymbol{c}) = d_i(l)\mathcal{N}(\boldsymbol{c}|\boldsymbol{\nu}_i, \boldsymbol{\eta}_i)\prod_{t=1}^{l}\mathcal{N}(\boldsymbol{o}_t|\boldsymbol{c}, \boldsymbol{\Sigma}_i). \tag{2.5}$$

The basic theory of GSHMMs is presented in (Russell 1993), including the extension of the

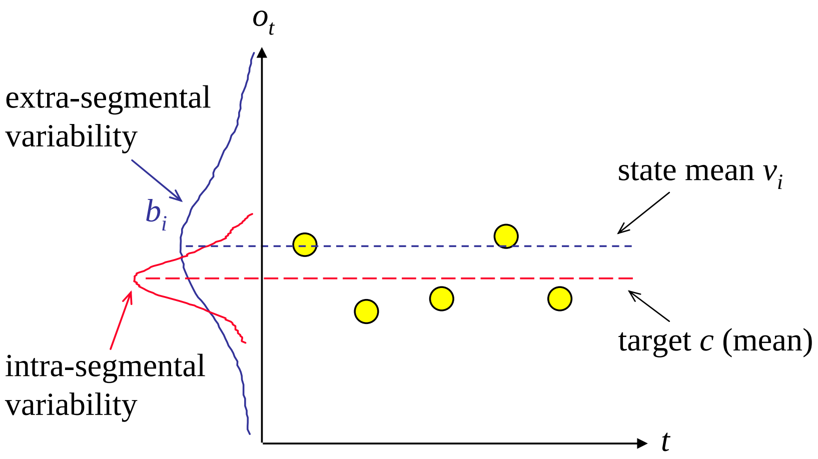


Figure 2.1: Separate modelling of extra- and intra-segmental variability in the static segmental HMM. Acoustic observations are assumed to be one-dimensional for simplicity. Figure adapted from (Holmes 2000).

conventional Baum-Welch parameter estimation algorithm to this type of model.

A similar approach was proposed by Gales & Young (1993), who extended the SHMM framework to include Gaussian mixture densities to model extra-segmental variations, and by Digalakis (1992) as a 'target state segment model', which is a special case of a dynamical system model, as described later in Section 2.3.8. The probability $p(\boldsymbol{o}_1^l|s_i)$ of the segment $\boldsymbol{o}_1^l$ given state s_i was calculated as the integral $p(\boldsymbol{o}_1^l|s_i) = \int_{\boldsymbol{c}} p(\boldsymbol{o}_1^l, \boldsymbol{c}|s_i) d\boldsymbol{c}$ in (Gales & Young 1993), whereas Russell (1993) proposed an estimation of the segment probability as

$$p(\boldsymbol{o}_1^l|s_i) \approx \hat{p}(\boldsymbol{o}_1^l|s_i) = p(\boldsymbol{o}_1^l, \hat{\boldsymbol{c}}|s_i), \tag{2.6}$$

based on the 'best' target $\hat{\boldsymbol{c}}$ which maximises the probability of $p(\boldsymbol{o}_1^l, \boldsymbol{c}|s_i)$

$$\hat{\boldsymbol{c}} = \arg\max_{\boldsymbol{c}} p(\boldsymbol{o}_1^l, \boldsymbol{c}|s_i). \tag{2.7}$$

In the case of a GSHMM, it can be found that the optimal target $\hat{\boldsymbol{c}}$ is given by

$$\hat{\boldsymbol{c}} = \frac{\boldsymbol{\nu}_i \boldsymbol{\Sigma}_i + \sum_{t=1}^{l} \boldsymbol{o}_t \boldsymbol{\eta}_i}{\boldsymbol{\Sigma}_i + l\boldsymbol{\eta}_i}. \tag{2.8}$$

That is, the optimal target $\hat{\boldsymbol{c}}$ is a weighted sum of the expected value and the actual acoustic observations. Gales & Young (1993) show that these two segment probability calculation approaches are related by

$$p(\boldsymbol{o}_1^l|s_i) = K\hat{p}(\boldsymbol{o}_1^l|s_i), \tag{2.9}$$

where K is a normalisation constant which is solely dependent on the segment length l and model variances, but not on the data.

Preliminary TIMIT results using CI SHMMs were disappointing, with an increased phone error rate (PER) relative to standard CI HMMs reported on a sub-set of the TIMIT corpus (Gales & Young 1993). Positive recognition results were achieved on a simpler task of connect digit recognition (Holmes & Russell 1995a). CD SHMMs were trained on either vocabulary-independent (i.e., training material is reading sentences) or vocabulary-dependent (i.e., training material is recording of four-digit strings) data, with 32% and 20% reduction in error rates respectively relative to standard HMMs. For CI SHMMs, recognition improvement relative to HMMs was only achieved when segmental models were trained on vocabulary-dependent data (27% reduction in error rates), but not on vocabulary-independent data. In addition, it was demonstrated that SHMMs provided a similar level of performance (except in the case of vocabulary-independent monophones) with fewer parameters when compared with two-component-mixture conventional HMMs. Later work (Holmes & Russell 1996) investigated the use of Gaussian mixtures to improve the intra-segmental model, and showed that the incorparation of a two-component Gaussian mixture intra-segmental distribution, where the two mixture components have the same mean but one has much smaller variance than the other, can significantly improve performance.

The use of two distributions is the key feature of the SHMM framework in attacking problems caused by the independence assumption. However, in practice, it is also the source of the principal difficulty, since imbalance between the extra- and intra-segmental probabilities leads to poor modelling and increased recognition errors (Holmes & Russell 1996). A static SHMM reduces the impact of the independence assumption of standard HMMs to some extent by imposing a continuity constraint which takes the form of a constant, random mean model of the speech segment. However, substantial performance improvements would not be expected until a proper model of dynamics is incorporated.

2.3.3 Linear probabilistic-trajectory segmental HMMs

In a linear trajectory segmental HMM (Holmes & Russell 1995*b*, Holmes & Russell 1997, Russell & Holmes 1997), the relationship between observations in a segment $\boldsymbol{o}_1^l = \{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_l\}$ is characterised by a 'noisy', linear trajectory, instead of a constant trajectory as in a static SHMM. A linear trajectory $\boldsymbol{f}_{(\boldsymbol{m},\boldsymbol{c})}$ of duration l can be defined as

$$\boldsymbol{f}(t) = (t - \bar{t})\boldsymbol{m} + \boldsymbol{c}, \tag{2.10}$$

where $\bar{t} = (l+1)/2$ is the midpoint of the start and end time points, and $\boldsymbol{m}$, $\boldsymbol{c}$ are the slope vector and mid-point vector respectively (whose dimension is that of the acoustic-feature space). Recall that the values $\boldsymbol{m}'$ and $\boldsymbol{c}'$ for which $\boldsymbol{f}_{(\boldsymbol{m},\boldsymbol{c})}$ best fits $\boldsymbol{o}_1^l$ (in a least-squares error sense) are given by

$$\boldsymbol{m}' = \frac{\sum_{t=1}^{l}(t-\bar{t})\boldsymbol{o}_t}{\sum_{t=1}^{l}(t-\bar{t})^2}, \quad \boldsymbol{c}' = \frac{\sum_{t=1}^{l}\boldsymbol{o}_t}{l}. \tag{2.11}$$

Intuitively, trajectories may be different for different utterances of the same sound because of extra-segmental factors, such as differences between speakers or chosen pronunciations for the speech sound. Therefore, it is unlikely to provide a very accurate representation for any one model unit by using a single trajectory to characterise all examples of that unit. One possible solution is to use a mixture of trajectories as described in (Gish & Ng 1993). However, many mixture components will be required to accommodate all possible trajectories. Alternatively, Holmes & Russell (1999) suggested a linear 'probabilistic-trajectory segmental HMM' (PTSHMM), in which both of the slope and mid-point vectors were assumed to be random variables with Gaussian distributions (assuming diagonal covariance matrices): $\boldsymbol{m} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\gamma})$, $\boldsymbol{c} \sim \mathcal{N}(\boldsymbol{\nu}, \boldsymbol{\eta})$. Due to the uncertainty of the trajectory parameters $\boldsymbol{m}$ and $\boldsymbol{c}$, the model in effect represents a continuous mixture of trajectories, each one providing a plausible time-evolving series of distribution means for any one segment example.

As in a static SHMM, individual observations within a segment are mutually independent for a given state s_i, though they are conditioned on the trajectory $\boldsymbol{f}$ associated with that state. In a linear PTSHMM, the joint probability of a sequence of acoustic observations $\boldsymbol{o}_1^l$ of length l and a particular linear trajectory $\boldsymbol{f}$ given state s_i is

$$\begin{aligned} p(\boldsymbol{o}_1^l, \boldsymbol{f}|s_i) &= d_i(l)p(\boldsymbol{f})\prod_{t=1}^{l} p\big(\boldsymbol{o}_t|\boldsymbol{f}(t)\big) \\ &= d_i(l)\mathcal{N}(\boldsymbol{m}|\boldsymbol{\mu}_i, \boldsymbol{\gamma}_i)\mathcal{N}(\boldsymbol{c}|\boldsymbol{\nu}_i, \boldsymbol{\eta}_i)\prod_{t=1}^{l}\mathcal{N}\big(\boldsymbol{o}_t|\boldsymbol{f}(t), \Sigma_i\big), \end{aligned} \tag{2.12}$$

where Σ_i is the fixed covariance matrix of the intra-segment Gaussian PDF.

Like a static SHMM, two approaches have been investigated to calculate the segment probability $p(\boldsymbol{o}_1^l|s_i)$ for a linear PTSHMM. One option adopted in early work (Holmes & Russell 1995*b*, Holmes & Russell 1997, Russell & Holmes 1997) is to approximate the segment probability as: $p(\boldsymbol{o}_1^l|s_i) \approx \hat{p}(\boldsymbol{o}_1^l|s_i) = p(\boldsymbol{o}_1^l, \hat{\boldsymbol{f}}|s_i)$, where $\hat{\boldsymbol{f}}$ is the *most-likely-trajectory* (Holmes & Russell 1999) which maximises the joint probability of the observations and the trajectory

$$\hat{\boldsymbol{f}} = \arg\max_{\boldsymbol{m},\boldsymbol{c}} p(\boldsymbol{o}_1^l, \boldsymbol{f}_{(\boldsymbol{m},\boldsymbol{c})}|s_i). \tag{2.13}$$

The most-likely-trajectory $\hat{\boldsymbol{f}}$ is defined by maximum *a posteriori* estimates of the slope ($\hat{\boldsymbol{m}}$) and mid-point ($\hat{\boldsymbol{c}}$), which are obtained by expanding $\log p(\boldsymbol{o}_1^l, \boldsymbol{f}|s_i)$ (Equations 2.12 and 2.10) according to the definition of a Gaussian distribution, differentiating with respect to $\boldsymbol{m}$ and $\boldsymbol{c}$, setting the partial derivatives to zero and solving

$$\hat{\boldsymbol{m}} = \frac{\left(\sum\limits_{t=1}^{l}(t-\bar{t})\boldsymbol{o}_t\right)\boldsymbol{\gamma}_i + \boldsymbol{\mu}_i\boldsymbol{\Sigma}_i}{\left(\sum\limits_{t=1}^{l}(t-\bar{t})^2\right)\boldsymbol{\gamma}_i + \boldsymbol{\Sigma}_i}, \tag{2.14}$$

$$\hat{\boldsymbol{c}} = \frac{\left(\sum\limits_{t=1}^{l}\boldsymbol{o}_t\right)\boldsymbol{\eta}_i + \boldsymbol{\nu}_i\boldsymbol{\Sigma}_i}{l\boldsymbol{\eta}_i + \boldsymbol{\Sigma}_i}. \tag{2.15}$$

Equations (2.14, 2.15) show that the most likely slope $\hat{\boldsymbol{m}}$ and mid-point $\hat{\boldsymbol{c}}$ are each a weighted sum of the values which are optimal with respect to the data and the expected values as defined by the model. Substituting $\hat{\boldsymbol{m}}$ and $\hat{\boldsymbol{c}}$ defined by Equations (2.14, 2.15) into equation $\hat{p}(\boldsymbol{o}_1^l|s_i) = p(\boldsymbol{o}_1^l, \hat{\boldsymbol{f}}_{(\boldsymbol{m},\boldsymbol{c})}|s_i)$ gives the most-likely-trajectory output probability $\hat{p}(\boldsymbol{o}_1^l|s_i)$.

An alternative approach, which was also used by Gales & Young (1993) for a static model, is what is referred to as the *trajectory-independent probability calculation* (Holmes & Russell 1999), in which all possible trajectories given state s_i are considered

$$p(\boldsymbol{o}_1^l|s_i) = \int_{\boldsymbol{f}} p(\boldsymbol{o}_1^l, \boldsymbol{f}|s_i) = d_i(l) \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \mathcal{N}(\boldsymbol{m}|\boldsymbol{\mu}_i, \boldsymbol{\gamma}_i) \mathcal{N}(\boldsymbol{c}|\boldsymbol{\nu}_i, \boldsymbol{\eta}_i) \times \prod_{t=1}^{l} \mathcal{N}\big(\boldsymbol{o}_t|\boldsymbol{f}(t), \boldsymbol{\Sigma}_i\big) d\boldsymbol{c}\, d\boldsymbol{m}. \tag{2.16}$$

The relationship between these two approaches to the segment probability calculation is given by $p(\boldsymbol{o}_1^l|s_i) = K\hat{p}(\boldsymbol{o}_1^l|s_i)$, where K is a normalising constant term which depends on the model variances and on segment duration, but not on the data (Holmes & Russell 1999).

Experiments exploring the consequences of the differences between the two approaches to computing output probabilities were reported in (Holmes & Russell 1999). Both static and linear PTSHMMs were evaluated on a small corpus (Russell, Moore, Tomlinson & Deacon 1983), which includes four lists of 50 connected-digit triples spoken by each of 10 male speakers (giving a total of 2000 words in the database). Experimental results showed that the trajectory-independent probability calculation method outperformed its most-likely-trajectory counterpart, with 75.6%, 71% and 60% reduction in WERs for static, linear (*flexible slope*) and linear (*constained slope*) PTSHMMs respectively.[6] The extent of the drop in performance when swtiching from the trajectory-independent approach to the most-likely-trajectory method demonstrates the difficulty in making *unbiased* trajectory estimates since the most-likely-trajectory is strongly infulenced by the data, and so the variance of the observations around that trajectory tends to be smaller than the variance around the 'true' tra-

[6] The flexible-slope approach determins the means and variances of the individual trajectory slopes based on collected statistics of all training examples, while the constrained-slope approach allows no variability in the slope varaiance parameters.

jectory, especially for short segments.

The fullest investigation into recognition using PTSHMMs was reported by Holmes & Russell (1999), who presented theoretical and experimental comparisons between different types of PTSHMMs, simpler SHMMs and conventional HMMs. However, before looking at the statistics, it is a good time to review a class of related models, which, in the terminology of Holmes & Russell (1999), is referred to as a *fixed-trajectory* segmental HMM, as opposed to a *probabilistic-trajectory* SHMM of their own work.

2.3.4 Fixed-trajectory segmental HMMs (FTSHMMs)

The linear trajectory models proposed by Gish & Ng (1993) (as a segmental model) and Deng, Aksmanovic, Sun & Wu (1994) (as a non-stationary or trended HMM) can be regarded as a limiting case of a linear PTSHMM (Holmes & Russell 1999) for which the extra-segment variances $\boldsymbol{\gamma}$ and $\boldsymbol{\eta}$ of the slope and mid-point vectors are both fixed at zero. Since there is no uncertainty in the trajectory parameters, a single fixed linear trajectory is associated with each state s_i, and the slope and mid-point values are always equal to the model means. In addition to constant and linear trajectories, higher order polynomial trajectories were also investigated, such as quadratic trajectories (Gish & Ng 1993, Deng et al. 1994) and cubic trajectories (Deng et al. 1994).

Gish & Ng (1993) suggested a segment model in which the time-varying mean trajectory of the feature vectors in a variable-length segment is represented by a polynomial. In such a model, an N-frame speech segment (each frame is a D dimensional vector) is modelled as

$$\boldsymbol{C} = \boldsymbol{Z}\boldsymbol{B} + \boldsymbol{E}, \tag{2.17}$$

where $\boldsymbol{C}$ is a $N \times D$ matrix representing the speech segment (each row vector corresponds to a frame), $\boldsymbol{Z}$ an $N \times R$ design matrix, $\boldsymbol{B}$ an $R \times D$ trajectory parameter matrix, and $\boldsymbol{E}$ a residual error matrix. R is the number of parameters in the polynomial function such that $R = 1$ for constant, $R = 2$ for linear and $R = 3$ for quadratic trajectories.

The role of the design matrix $\boldsymbol{Z}$ is to perform duration normalisation for each segment so that its frames are distributed uniformly between times 0 and 1. Taking $R = 3$ as an example, where a quadratic trajectory is specified, $\boldsymbol{Z}$ becomes

$$\boldsymbol{Z} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & \frac{1}{N-1} & \left(\frac{1}{N-1}\right)^2 \\ 1 & \frac{2}{N-1} & \left(\frac{2}{N-1}\right)^2 \\ \vdots & \vdots & \vdots \\ 1 & 1 & 1 \end{bmatrix}. \tag{2.18}$$

In this case, the i^{th} feature dimension of the segment (i.e., the i^{th} column of $\boldsymbol{C}$) is

$$\begin{bmatrix} c_{1,i} \\ c_{2,i} \\ \vdots \\ c_{N,i} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & \frac{1}{N-1} & \left(\frac{1}{N-1}\right)^2 \\ \vdots & \vdots & \vdots \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} b_{1,i} \\ b_{2,i} \\ b_{3,i} \end{bmatrix} + \begin{bmatrix} e_{1,i} \\ e_{2,i} \\ \vdots \\ e_{N,i} \end{bmatrix}, \tag{2.19}$$

or

$$c_{n,i} = b_{1,i} + b_{2,i}\Big(\frac{n-1}{N-1}\Big) + b_{3,i}\Big(\frac{n-1}{N-1}\Big)^2 + e_{n,i}, \tag{2.20}$$

where $n = 1, \ldots, N$ and $i = 1, \ldots, D$.

Instead of using normalised time to describe the polynomial trajectories, as is shown in Equation (2.20), Deng et al. (1994) employed absolute time to specify the polynomial regression functions. Assume that the segment $\boldsymbol{C}$ is associated with state j, then Equation (2.20) becomes

$$c_{n,i} = b_{1,i} + b_{2,i}(t - t_j) + b_{3,i}(t - t_j)^2 + e_{n,i}, \tag{2.21}$$

where t_j is an auxiliary parameter which registers the time when state j in the HMM is just entered so that $(t - t_j) \propto n$ for the nth frame in the segment. Using absolute time has the advantage of efficient recognition and segmentation algorithms since the Markov assumption holds within and across segments (Ostendorf et al. 1996), while phonetic alignments must be available or derived by some other means with the model described by Gish & Ng (1993).

Given a segment k (of length N_k) with data matrix $\boldsymbol{C}_k$ and design matrix $\boldsymbol{Z}_k$, the maximum likelihood estimates of the trajectory parameters are given by using the least squares method

$$\hat{\boldsymbol{B}}_k = [\boldsymbol{Z}_k^\top \boldsymbol{Z}_k]^{-1} \boldsymbol{Z}_k^\top \boldsymbol{C}_k, \tag{2.22}$$

where $^\top$ and $^{-1}$ denote the transpose and inverse respectively. In both (Gish & Ng 1993) and (Deng et al. 1994), the residual errors are assumed to be IID from frame to frame with a zero-mean Gaussian distribution $\mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_k)$, where $\boldsymbol{\Sigma}_k$ is the state-dependent covariance matrix identical for all frames in the segment k. Once the trajectory parameter matrix $\hat{\boldsymbol{B}}_k$ is estimated, the residual error covariance matrix for the segment, $\hat{\boldsymbol{\Sigma}}_k$, is given by

$$\hat{\boldsymbol{\Sigma}}_k = \frac{\hat{\boldsymbol{E}}_k^\top \hat{\boldsymbol{E}}_k}{N_k} = \frac{\left(\boldsymbol{C}_k - \boldsymbol{Z}_k \hat{\boldsymbol{B}}_k\right)^\top \left(\boldsymbol{C}_k - \boldsymbol{Z}_k \hat{\boldsymbol{B}}_k\right)}{N_k}, \tag{2.23}$$

where N_k is the number of frames in segment k.

Deng et al. (1994) conducted a quantitative analysis on the effect of different trajectories (including piecewise constant, linear, quadratic and cubic trended functions) by fitting actual speech data (both the training data and the test data, in the form of mel-cepstral coefficients). Firstly, the optimal segmentation of the data was found via a modified Viterbi algorithm. Then various polynomial functions were fitted to the segmented data according to the corresponding states. The measure for the accuracy of the data fitting was obtained by summing state-dependent frame residual errors over frames, with a smaller value indicating better performance. Experiment results showed that the data fitting error decreased quickly as the polynomial order increased. These results provided some evidence that the use of higher order trajectories could be beneficial in improving recognition performance.

Experiments on a TIMIT speaker-independent vowel classification task (including 13 monothongs and 3 diphthongs)[7] using segment models were conducted by Gish & Ng (1993). Later, Gish & Ng (1996) evaluated more complex trajectory models with Gaussian mixtures and time-varying covariances on the same task. The main findings are: 1) the use of higher

[7] The 13 monothongs are [iy, ih, ey, eh, ae, aa, ah, ao, ow, uw, uh, ux, er] and the 3 diphthongs are [ay, oy, aw]. The phonetic symbols used here follow the TIMIT labelling convention.

order polynomials (from constant to linear and finally to quadratic) leads to a steady, although moderate, improvement in recognition performance; 2) only some vowels, mainly the diphthongs, require quadratic trajectories. In other words, linear trajectories are sufficient to characterise most sounds; 3) The use of Gaussian mixture models and time-varying covariances results in further improvements in performance. Similar results were reported by Deng et al. (1994) on an isolated-word speaker-dependent 36-CVC-word[8] speech recognition experiment.

Based on the work of Gish & Ng, Yun & Oh (2002) developed a full continuous speech recognition system. Instead of modelling variable-length speech segments as in previous work (Gish & Ng 1993, Gish & Ng 1996), Yun & Oh fitted fixed-length sliding windows on the acoustic observations. The design matrix $\boldsymbol{Z}$ was modified accordingly to centre on the current observation vector, and therefore frames within a segment were distributed uniformly between normalised times $[-0.5, +0.5]$. For example, given a speech segment of length $N = 2M + 1$, the design matrix for a quadratic trajectory becomes

$$\boldsymbol{Z} = \begin{bmatrix} 1 & -\frac{M}{2M} & -\left(\frac{M}{2M}\right)^2 \\ 1 & -\frac{M-1}{2M} & -\left(\frac{M-1}{2M}\right)^2 \\ \vdots & \vdots & \vdots \\ 1 & 0 & 0 \\ \vdots & \vdots & \vdots \\ 1 & \frac{M-1}{2M} & \left(\frac{M-1}{2M}\right)^2 \\ 1 & \frac{M}{2M} & \left(\frac{M}{2M}\right)^2 \end{bmatrix} . \tag{2.24}$$

Yun & Oh (2002) conducted TIMIT phone recognition experiments and compared the performance of the new segmental-feature HMM (SFHMM) with that of conventional HMMs. Only CI models were evaluated (including 48 monophone models), and a phone bigram language model was also used in their experiments. Using 12-MFCC plus log energy, the SFHMM gave a slight improvement over conventional HMMs which also included Δ fea-

[8] These 36 CVC words, in which ‘C’ encompasses six stop consonants [p, t, k, b, d, g] and ‘V’ is the vowel [i:], were spoken with a short pause in between by three native English speakers in a normal office environment.

tures, provided that the segment length was set at the value of 5 and the polynomial order was greater than 3. When SFHMMs were used with static and Δ features, the new model outperformed conventional HMMs which included both Δ and Δ^2 parameters. For instance, the standard HMM system with 2 Gaussian mixture components gave a phone accuracy of 57.0%, while an SFHMM with fixed segment length of 5 and polynomial order of 3 produced an accuracy of 60.1%.

2.3.5 PTSHMMs summary

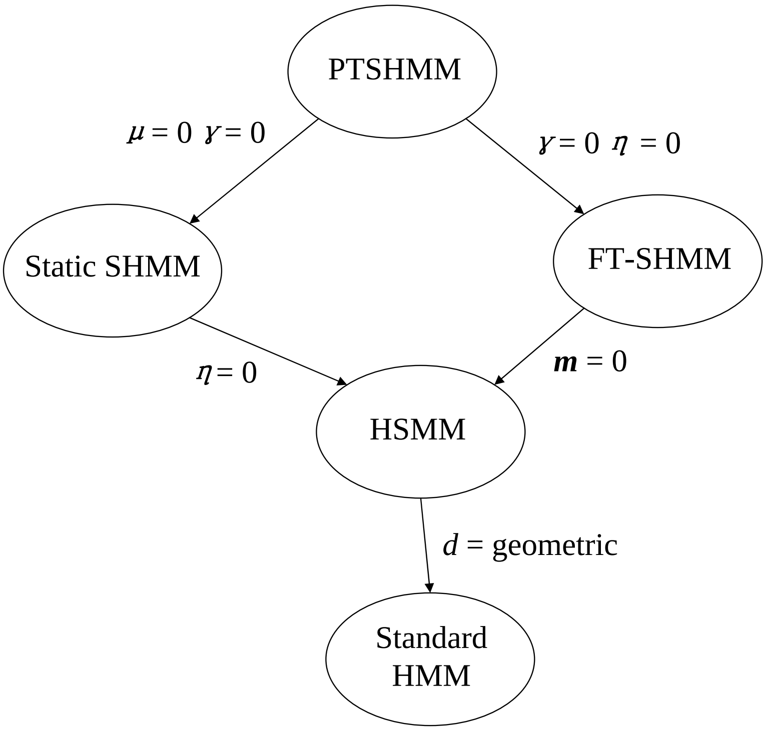


Figure 2.2: The family tree of PTSHMMs. This figure shows how a linear PTSHMM can be related to other segmental models by making certain simplifying assumptions. Each line indicates that there is a direct link between the two types of models it connects, with the arrow pointing to the simplified model under the conditions given beside the line.

Many acoustic models reviewed so far in this chapter can be regarded as special cases of a PTSHMM. Figure 2.2 shows the relationships between models of the PTSHMM family. For example, a static SHMM (Russell 1993, Gales & Young 1993) can be seen as a limiting case of a linear PTSHMM for which both the slope mean μ and the slope variance γ are fixed at zero, giving a constant trajectory defined by the mid-point parameter. An HSMM can be viewed as a limiting case of a static SHMM for which the mid-point variance η is zero, or as a limiting case of a linear FTSHMM for which the slope m is fixed at zero.

Holmes & Russell (1999) compared phone classification results on the male portion of the TIMIT core test set for different sets of segment-based models shown in Figure 2.2. Both CI (60-symbol set) and right-CD models were used. The first 12 MFCCs together with an average amplitude feature formed the basic feature set for their experimental work. In addition, dynamic features were also used in some experiments. All the PTSHMM experiments were carried out using the trajectory-independent approach to computing segment probabilities.

Model set	%Errors	
	MFCCs only	MFCCs+Δ features
HMMs	40.9	29.8
HSMMs	43.0	29.4
Static PTSHMMs	45.0	32.2
Linear FTSHMMs	39.0	27.4
Linear PTSHMMs	38.2	26.8

Table 2.1: This table highlights main phone classification results (data taken from (Holmes & Russell 1999)) for different types of CD PTSHMMs, compared with standard HMMs, on the core test set (male only) of the TIMIT corpus.

Experimental results using CD models are summarised in Table 2.1. HSMMs performed very slightly better than HMMs when time-derivatives were included, but gave somewhat worse performance than HMMs when only static features were used. The difference in HSMMs performance demonstrates the importance of modelling feature dynamics, as the use of Δ features provides a means of incorporating dynamic information in acoustic modelling. The use of static PTSHMMs did not lead to classification improvements over conventional HMMs. Linear PTSHMMs outperformed linear FTSHMMs, which in turn outperformed conventional HMMs, with or without time-derivatives. These results showed the advantages of introducing a linear trajectory model, and further benefits when a distinction is made between extra- and intra-segmental variability in conjunction with the linear trajectory model. Experiments results using CI models followed the same pattern.

2.3.6 Non-parametric segmental HMMs

Previous sections have reviewed typical trajectory-based SHMMs in which the segment mean is specified by a constant, linear or higher order polynomial trajectory. An alternative approach to modelling acoustic feature dynamics takes the form of a non-parametric trajectory model, in which the sequence of distributions for a segment (i.e., the mean trajectory) is represented by a non-parametric trajectory, and each distribution mean corresponds directly with a point along the trajectory.

Motivated by the fact that speech is really produced by a non-stationary process, Ghitza & Sondhi (1993) transformed the standard HMM into a non-stationary HMM, in which each HMM state is defined as a single, fixed-length *template*. Instead of phones as used in many other segment models, diphones (i.e., transitions between a pair of adjacent phones) were used as the modelling units in Ghitza & Sondhi's model. Let $\bar{\boldsymbol{O}}$ be the template of length $\bar{T}$ (or a 'typical' sequence of $\bar{T}$ observations) for a given HMM state s_i and $\mathcal{O} = \{\boldsymbol{O}_1, \ldots, \boldsymbol{O}_K\}$ be the K observation segments in the database corresponding to that state. Following the modified k-means clustering method (Wilpon & Rabiner 1985), the template $\bar{\boldsymbol{O}}$ (i.e., the sequence of Gaussian distribution means for state s_i) is specified as the observation sequence in the set $\mathcal{O}$ whose cumulative distance from all other observation sequences in the ensemble $\mathcal{O}$ is a minimum

$$\sum_{k=1}^{K} D(\bar{\boldsymbol{O}}, \boldsymbol{O}_k) \leqslant \sum_{k=1}^{K} D(\boldsymbol{O}_j, \boldsymbol{O}_k), \tag{2.25}$$

for all j, where $j = 1, \ldots, K$, and $D(\boldsymbol{O}_j, \boldsymbol{O}_k)$ denotes the distance between the two observation sequences $\boldsymbol{O}_j$ and $\boldsymbol{O}_k$, which is calculated by the usual dynamic time warping (DTW) procedure (see, e.g., Holmes & Holmes (2001)).

A similar non-parametric trajectory model, known as the 'stochastic segment model' (SSM) in the literature, was described in (Ostendorf & Roukos 1989). The SSM models the sequence of observation vectors over the entire speech segment of a phone. The main difference between these two non-parametric approaches is that a deterministic time-warping transformation (either a linear time sampling or a space sampling) was used in SSMs to transform variable-length segments to the fixed-length template, whereas dynamic time warping based on dynamic programming was used by Ghitza & Sondhi (1993). Deterministic mappings have the advantage of reduced computation relative to dynamic programming, and for phone-sized units and smaller, they work quite well in practice (Ostendorf et al. 1996). In both approaches, a block-diagonal covariance structure (where each block corresponds to a sample covariance) was used to reduce the parameterization by assuming that successive frames within a segment are independent given the segment length.

Using non-parametric segmental models, both authors reported positive experimental results compared with conventional HMMs. For example, speaker-dependent phoneme recognition experiment results presented in (Ghitza & Sondhi 1993) showed that the new segmental HMM outperformed the base-line HMM system (Levinson 1986), yielding $20 - 30\%$ reductions in PERs. Ostendorf & Roukos (1989) built a word recognition system based on SMMs, which reduced WERs by approximately one-third in comparison to conventional HMMs on a speaker-dependent, CI continuous speech recognition task.

2.3.7 Comparison of parametric and non-parametric models

Sections 2.3.2 – 2.3.5 discuss various parametric SHMMs and Section 2.3.6 reviews typical non-parametric models. The parametric and non-parametric approaches each have their respective advantages and disadvantages. A comparison between these two approaches has been provided by Kannan & Ostendorf (1998) in the context of LVCSR using the Wall Street Journal (WSJ) (Paul & Baker 1992) and Switchboard corpora.

From the perspective of speech pattern modelling, parametric models are better suited for certain classes of speech, such as vowels or sub-phone units, which often exhibit smooth trajectories. For those units that do not vary smoothly in time (e.g., stop consonants), multiple segments can be used. On the other hand, non-parametric models are more suitable for modelling phones including discontinuities which can not be simply represented by parametric trajectories, e.g., in stops between the closure and the burst. Parametric models generally have a more compact parameterization than non-parametric models, which can be potentially advantageous when limited training and/or adaptation data is available. However, non-parametric approaches offer some computational (and/or storage) advantages since distribution means can be stored in a small table and score caching can be used for reducing computation (Ostendorf et al. 1996).

In terms of recognition performance, the N-best list rescoring experiments on either WSJ or Switchboard conducted by Kannan & Ostendorf (1998) showed that the parametric approach (using two linear trajectories per phone, which can be thought of as approximating a quadratic trajectory) provided similar, but slightly lower performance than the non-parametric approach (using eight regions per phone). For example, the non-parametric system gave WERs of 13.2% and 45.6%, compared to 13.6% and 46.8% given by the parametric system, on WSJ and Swithchboard respectively. The likely explanation for the slight degradation for the parametric system may be because of the constant covariance assumption used in the parametric model but not in the non-parametric model. This was also supported by Gish & Ng (1996), who showed that relaxing the constant covariance assumption for parametric models resulted in improved recognition performance.

2.3.8 Linear dynamical systems

A stochastic, linear dynamical system (LDS) was first introduced as a speech recognition model by Digalakis et al. (Digalakis 1992, Digalakis, Rohlicek & Ostendorf 1993). The application of the LDS for ASR has recently been readdressed by Frankel et al. (Frankel & King 2001, Frankel 2003, Frankel & King 2007), referred to as a linear dynamic model (LDM), to describe articulator movement at the University of Edinburgh. A linear dynamical system is, in general, identified by the following pair of equations:

$$\boldsymbol{x}_t = \boldsymbol{F}\boldsymbol{x}_{t-1} + \boldsymbol{w}_t, \tag{2.26}$$

$$\boldsymbol{o}_t = \boldsymbol{H}\boldsymbol{x}_t + \boldsymbol{v}_t, \tag{2.27}$$

where $\boldsymbol{x}_t$ is an m-dimensional unobserved state vector, and $\boldsymbol{o}_t$ is an n-dimensional observation vector. The underlying dynamical *state process* of an LDS is specified by Equation (2.26), in which state propagation is governed by a linear transformation via the $m \times m$ state evolution matrix $\boldsymbol{F}$ and the addition of Gaussian noise $\boldsymbol{w}_t \sim \mathcal{N}(\boldsymbol{\mu}^{(x)}, \boldsymbol{\Sigma}^{(x)})$. The *observation process* is described by Equation (2.27), which is linked to the state process via a linear projection $\boldsymbol{H}$ (the $n \times m$ observation matrix) with the addition of observation noise $\boldsymbol{v}_t \sim \mathcal{N}(\boldsymbol{\mu}^{(o)}, \boldsymbol{\Sigma}^{(o)})$. The two noise vectors $\boldsymbol{w}_t$ and $\boldsymbol{v}_t$ are assumed to be independent. Furthermore, the initial state $\boldsymbol{x}_0$ is also Gaussian: $\boldsymbol{x}_0 \sim \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$.

Many statistical acoustic models can be regarded as special cases of an LDS model, and the relationships between the dynamical system segment model and various acoustic models was discussed in (Ostendorf et al. 1996). For example, the static SHMM proposed by Russell (1993) and Gales & Young (1993) (see Section 2.3.2) assumes a constant mean throughout the segment $\boldsymbol{o}_t \sim \mathcal{N}(\boldsymbol{c}, \boldsymbol{\Sigma}_i)$, where the mean $\boldsymbol{c}$ is modelled by a Gaussian prior $\boldsymbol{c} \sim \mathcal{N}(\boldsymbol{\nu}_i, \boldsymbol{\eta}_i)$. This static model can be seen as the *target state* (Digalakis 1992) version of an LDS model, in which the hidden state is fixed at its initial value for all t, $\boldsymbol{x}_t = \boldsymbol{x}_0 = \boldsymbol{c}$, i.e., $\boldsymbol{F} = \boldsymbol{I}$[9], $\boldsymbol{w}_t = \boldsymbol{0}$, $\boldsymbol{\mu}^{(o)} = \boldsymbol{0}$ and $\boldsymbol{\Sigma}^{(o)} = \boldsymbol{\Sigma}_i$. The distribution of the target state represents the more global forms of variability, such as speaker identity, and the target state, once chosen, remains fixed throughout the entire segment.

In the case that the unobserved state $\boldsymbol{x}_t$ is taken to be zero, the LDS model is then fully defined by the term $\boldsymbol{v}_t$; that is, $\boldsymbol{o}_t = \boldsymbol{v}_t$. This special type of LDS model corresponds to the constrained-mean trajectory assumption, which includes examples of both parametric mean trajectory models (e.g., the fixed-trajectory segmental HMM described in Section 2.3.4) and non-parametric mean trajectory models (such as the non-stationary HMM and the stochastic segment model described in Section 2.3.6), depending on the definition of the term $\boldsymbol{v}_t$.

In the original application of LDS by Digalakis (1992), the dimension of the state vector is assumed to be equal to the size of the observation vector, i.e., $m = n$, the observation matrix $\boldsymbol{H} = \boldsymbol{I}$ for all models and the observation noise covariance matrix $\boldsymbol{\Sigma}^{(o)}$ shared globally by all models. As a result, the observation process is assumed to be a noisy version of an underlying Gauss-Markov process (defined by Equation 2.26) (Wellekens 1987, Kenny, Lennig & Mermelstein 1990), which explicitly models correlation between neighbouring

[9] $\boldsymbol{I}$ is used to denote an identity matrix.

feature vectors within a segment $\boldsymbol{o}_1^l = \{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_l\}$

$$b_i(\boldsymbol{o}_1^l) = \prod_{t=1}^{l} p(\boldsymbol{o}_t | \boldsymbol{o}_{t-1}, s_i). \tag{2.28}$$

Therefore, the LDS segment model includes the Gauss-Markov process as a special case, for which $\boldsymbol{H} = \boldsymbol{I}, \boldsymbol{v}_t = \boldsymbol{0}$ and $\boldsymbol{o}_t = \boldsymbol{x}_t$.

Digalakis (1992) described a model parameter estimation algorithm based on the EM technique and a recognition algorithm for LDS models. CI phone classification experiments on TIMIT showed that LDS models outperformed other segment models, including the independent-frame model (i.e., the block-diagonal SSM), the target state model and the Gauss-Markov model, with $9 - 15\%$ reduction in error rate relative to other models based on the 18 cepstra feature set. When feature derivatives were used, the LDS model gave the highest accuracy of 73.9%, while the independent-frame SSM gave an accuracy of 72.1%. In phone recognition experiments, the LDS segment model also gave better performance than the independent-frame model, with 58% and 55% accuracies, respectively, on static 18 cepsetra coefficients, though a similar level of performance was achieved (63% and 64% for the LDS model and the independent-frame model, respectively) when the first time derivatives were also included in the acoustic features.

2.4 Hidden dynamic models

Section 2.3 reviews a variety of SHMMs appeared in the literature. These approaches intend to explicitly model dependencies between speech vectors within a variable-length segment in the *acoustic domain*, thus mitigating problems caused by the independence assumption in conventional HMMs. Most of these models register some gains in recognition performance compared with HMMs on standard tests. However, one of the disadvantages of these approaches is the unsuitability of a spectrum-based acoustic representation for modelling dynamics, in which simple articulator movements often exhibit complex changes. Moreover, in SHMMs, successive segments are assumed to be independent. It is believed that due to

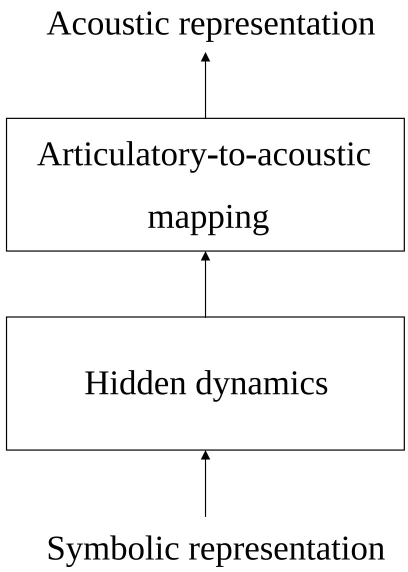


Figure 2.3: A hidden dynamic model in which speech dynamics are modelled in an intermediate layer and transformed to the acoustic layer via an articulatory-to-acoustic mapping.

strong coarticulation in speech production, this assumption is not valid. HDMs, as shown in Figure 2.3, attempt to overcome these limitations of acoustic dynamic models by incorporating an intermediate layer (referred to as a hidden layer since it is not accessed directly from the acoustics) in which dynamics can be modelled directly in an articulatory-related space. A major theoretical advantage of such an approach is that coarticulation can be dealt with naturally and efficiently in a potentially low-dimensional, articulatory domain (i.e., the hidden layer). By exploiting speech production knowledge at this deeper, production level, HDMs may be able to capture additional information about acoustic-phonetic relationships which is unavailable in current acoustic models.

Early work on HDMs was due to Bakis (1991), who proposed a very general model with targets, linear dynamics and non-linear output mapping. This model was rediscovered later by Deng (1998) and Richards & Bridle (1999). These two approaches were investigated, evaluated and compared at the 1998 Workshop on Language Engineering, Johns Hopkins University (Picone, Pike, Reagan, Kamm, Bridle, Deng, Ma, Richards & Schuster 1999). Deng, Yu & Acero (2006) provide a more recent review on many different types of hidden dynamic speech models.

Before describing the HDM framework and various HDM implementations, it is useful to revisit the state-space model (see, e.g., Rosti (2004)). Let $\boldsymbol{x}_t$ and $\boldsymbol{o}_t$ be an m-dimensional state vector and an n-dimensional observation vector, respectively. A general state-space model is described by the following two equations:

$$\boldsymbol{x}_t = f(\boldsymbol{x}_{t-1}, \ldots, \boldsymbol{x}_1, \boldsymbol{w}_t), \tag{2.29}$$

$$\boldsymbol{o}_t = h(\boldsymbol{x}_t, \boldsymbol{v}_t), \tag{2.30}$$

where the state evolution is defined by $f(\cdot)$ and transformed to the observation space via the mapping $h(\cdot)$. $\boldsymbol{w}_t$ and $\boldsymbol{v}_t$ are the state noise and observation noise respectively. A linear Gaussian model (Roweis & Ghahramani 1999, Rosti 2004) is a special case of the above general state-space model, in which both functions $f(\cdot)$ and $h(\cdot)$ are restricted to being linear transformations, and the noise terms $\boldsymbol{w}_t$ and $\boldsymbol{v}_t$ are assumed to be Gaussian. Apparently, the LDS model discussed in Section 2.3.8 is an example of a linear Gaussian model with a first-order state process.

The hidden dynamic model (as shown in Figure 2.3) can now be properly interpreted in terms of a state-space model. Intuitively, the state vector $\boldsymbol{x}_t$ represents formants or articulator positions at time t, and the state evolution function $f(\cdot)$ describes formant transitions or articulator movements in time. The state noise $\boldsymbol{w}_t$ characterises variations in the underlying production system, such as differences between speakers or speaking styles. The observation mapping $h(\cdot)$ transforms the underlying articulatory dynamics, along with the observation noise $\boldsymbol{v}_t$ onto the acoustic representation, where $\boldsymbol{v}_t$ models all the external sources of noise such as background noise or channel variability.

The rest of this section first describes an example system of hidden dynamic models proposed by Richards & Bridle (1999) (Section 2.4.1), and then discusses the HDM framework component by component (Sections 2.4.2–2.4.4), pointing out differences and similarities among a variety of HDM implementations. The main concern is, of course, about how the hidden dynamics are modelled in the hidden space (Section 2.4.3).

2.4.1 An example HDM system

Inspired by the classic HMS synthesis-by-rule system (Holmes et al. 1964) (which converts phone-segment sequences to formant patterns and hence waveforms, see Section 3.2.4), Richards & Bridle (1999) suggested a segmental HDM, which aims at explicitly accounting for the coarticulation and transitions between neighbouring phone segments in an utterance. Under such a model, each phone segment in the inventory is associated with a single target vector in the hidden dynamic space, which remains constant throughout the entire phone duration. The target for each phone segment can be thought of as corresponding to formant frequencies or articulator positions for that phone, though in (Richards & Bridle 1999) targets and transitions are more abstract than formant frequencies and amplitudes. An acoustic pattern is then assumed to be produced from a target sequence in the hidden dynamic space with given durations. (Note that the target is typically multi-dimensional.) In addition, each phone class is also associated with a vector of time-constant, which represents target importance weight or variance associated with that target.

Given a time-aligned phone sequence, the corresponding piecewise constant target sequence in the hidden space is smoothed using a zero-phase second-order low-pass filter (a simple Kalman smoother) (Kalman 1960, Haykin 2001) to generate a continuous trajectory through the hidden space. The variable low-pass filter is symmetrical so that the center of transitions occurs at phone boundaries, to agree with normal phonetic marking practice (Picone et al. 1999). The smoothed hidden trajectory is then transformed to the surface acoustic space via a single non-linear mapping for all phone classes in the form of an MLP. In (Bridle, Deng, Picone, Richards, Ma, Kamm, Schuster, Pike & Reagan 1999), this type of HDM is also referred to as a *deterministic* HDM (DHDM) since the whole production process is treated as deterministic until the acoustic output is generated, and Gaussian noise is then added to this acoustic pattern to model the variability.

All targets were initialized at zero, and the MLP had 40 hidden units in one hidden layer, whose weights were initialized with small random values. Model parameters estimation was by means of gradient descent (see, e.g., Bishop (1995)) on the difference E (using a

Euclidean distance) between the synthetic patterns and training data. Derivatives of the error E can be backpropagated through the MLP and through the filter to the targets, so that derivatives of E with respect to the MLP weights and the two phone-dependent parameters (i.e., the target and the time-constant) can be obtained.

Picone et al. (1999) reported N-best list rescoring experiment results on a subset of the Switchboard conversational speech corpus (including 1241 utterances, spoken by 23 male speakers). When the N-best list (where $N = 5$) contained reference transcriptions, the HDM system demonstrated superior performance to conventional HMMs, with 32% reduction in WER. In case the reference transcripiton was not included in the N-best list (either $N = 5$ or $N = 100$), the HDM system gave similar, though slightly inferior performance to conventional HMMs. It should be noted that the HDM system was trained only on a limited amount of speech data from a single speaker of the Switchboard corpus, whereas the HMM system was trained on a much larger training set. These results suggest that HDMs are capable of capturing extra information that conventional HMMs are incapable of. In addition, Picone et al. (1999) showed that the two HDM variants, i.e., the DHDM system (Richards & Bridle 1999) and the VTR system (Deng 1998), were fairly closely in performance.

2.4.2 Alternative symbolic representations of speech

In most ASR work, speech at the symbolic (phonetic) level is represented as a sequence of phonemes, oft-cited as the “beads-on-the-string” model of speech. Using phoneme as the basic unit for speech recognition has many theoretical and practical advantages. Firstly, any word in a language can be transcribed in terms of a sequence of phonemes of that language. Secondly, the total number of phonemes in any language is relatively small in comparison with the total number of words of that language. In English, for example, approximately 50 phonemes are sufficient to describe any word, while a typical state-of-the-art LVCSR system is expected to be able to cope with a large vocabulary over 65,000 words. Thirdly, there is a rich source of phoneme-based pronunciation dictionaries, which enable idealised pronunciations of most words in a language to be transcribed easily as a sequence of phonemes

(Russell 2003).

The "beads-on-the-string" view of speech leads to a naive, sequential model of speech pattern structure. Although this is a reasonable approach in many speech technologies, it is at variance with modern theories in speech science such as non-linear phonology, which treat speech in terms of multi-dimensional asynchronous processes of a set of distinctive features[10]. In recent year, many alternative approaches have been studied to represent speech at the symbolic level, among which feature-based approaches have shown some advantages over the traditional phone-based speech model (Gutkin & King 2004, Deng 1998).

Deng (1998) adopted a feature-based phonological model (or pronunciation model) in which speech was represented as multi-dimensional, asynchronous and overlapping articulatory features. A detailed description of the feature-based phonological model was given in (Deng & Sun 1994) and a brief review is provided here for completeness.

A total of five multi-valued features were used in (Deng & Sun 1994), including lips, tongue blade, tongue dorsum, velum and larynx. A detailed feature specification system was created, in which each phone corresponds to a distinct, static (hence context-independent) configuration of articulatory features (affricates and diphthongs, on the other hand, were decomposed into their associated relative stationary sub-segment, i.e., affricates → stop and fricative positions; diphthongs → concatenation of two target vowels). To account for context dependencies, features were allowed to spread in both left and right directions with a varying degree of spread depth, resulting in temporal overlap among different features. Given the feature-overlapping pattern, a state sequence was constructed in which each state (or the feature bundle) must be identified with a permissible feature group that can be found at a fixed time point in the feature-overlapping pattern.

Phonetic classification results on TIMIT showed that feature-based HMMs outperformed phonemic CI HMMs, with at most a 27% reduction in the error rate. Moreover, the superior performance of feature-based approach can be achieved with far fewer mixture components than conventional HMMs, which suggests that less traning data is required in the feature-

[10] The term 'feature' here is used to denote the symbolic indicators of phonetic contrasts, rather than acoustic observations as it is usually used to refer to in ASR.

based approach.

Investigating alternative pronunciation models is out of the scope of this thesis. Ostendorf (1999) provided some useful insights into non-standard strategies to represent speech other than phonemes.

2.4.3 Hidden dynamics modelling

Choice of the hidden representation

A choice must be made at the first place about the nature of the hidden representation in an HDM. This does not have to be a real articulatory description, though in this thesis it is often referred to as an articulatory layer. The vocal-tract-resonances (Deng 1998, Deng & Ma 2000, Zhou, Seide & Deng 2003) or formant frequencies (Russell & Jackson 2005, Russell et al. 2007), among others, provide an appropriate hidden representation because formants are closely related to not only articulation but also acoustics. VTRs are the pole locations of the vocal tract configured to produce all speech sounds, and have acoustic correlates of formants which are directly measurable for vocalic sounds, but often are hidden or perturbed for consonantal sounds due to the concurrent spectral zeros and turbulence noises (Deng & Ma 2000). For nonnasalized vowels, VTRs coincide with formants.

Apart from VTRs or formants, a number of alternative hidden representations have also been investigated in the past. For example, the intermediate representation in the HDM described in (Richards & Bridle 1999) is an unconstrained hidden space, which is more abstract than formant frequencies. In the coarticulation model presented in (Gao, Bakis, Huang & Zhang 2000), the hidden space is based on the five articulatory features as described in (Deng & Braam 1994), i.e., lips, tongue blade, tongue dorsum, velum and larynx.

The dimensionality of the hidden space varies, depending on the nature of the hidden representation. For example, when the hidden representation is based on formants, the first three formant frequencies are typically used, resulting in a three-dimensional hidden space (Deng 1998, Russell & Jackson 2005). Russell & Jackson (2005) also investigated two alternative hidden representations based on formant frequencies and amplitudes, which could lead up to a 12-dimensional hidden space. A higher-dimensional hidden space provides more information about the underlying production mechanisms, though leads to increased parameterisation and computation. Bridle et al. (1999) found in an experiment that two dimensions seemed adequate to reproduce certain vowel sounds.

Form of hidden dynamics

The hidden dynamics have been modelled in a variety of ways. Table 2.2 summarises various forms of hidden dynamics proposed in the literature. For notational simplicity, it is assumed that the hidden space is one-dimensional, and all unit (state or phone) indices are omitted for illustrative clarity.

Author	Form of hidden dynamics/State equation
Russell & Jackson (2005) Russell et al. (2007)	Piecewise linear $f(t) = (t - \bar{t})m + c$
Richards & Bridle (1999) Gao et al. (2000)	Smoothed piecewise constant (Kalman smoother) $(\sigma^{-1} + 2)x(t) = \sigma^{-1}c + x(t+1) + x(t-1)$
Deng (1998) Zhou et al. (2003) Seide et al. (2003)	Causal second-order low-pass filter target-direct trajectory function $x(t) = 2\rho x(t-1) - \rho^2 x(t-2) + (1-\rho)^2 c + w(t)$
Deng & Ma (2000)	Causal first-order low-pass filter $x(t) = 2\rho x(t-1) + (1-\rho)^2 c + w(t)$

Table 2.2: Summary of various forms of hidden dynamics appeared in the literature.

Piecewise linear The simplest form of dynamics is a piecewise linear trajectory model (Russell & Jackson 2005, Russell et al. 2007). Each state of the linear/linear MSHMM is associated with a variable-duration linear trajectory f in the formant-based intermediate layer, and therefore formant dynamics are modelled as piecewise linear trajectories. m and c are the slope and mid-point of the linear trajectory f and $\bar{t} = (l+1)/2$ where l is the trajectory duration. Although such an approach offers computational advantages, the lack of continuity constraints at segment boundaries poses as a major drawback because the temporal smoothness of formant transitions between speech units is lost. The research presented in this thesis attempts to overcome this limitation by using a continuous, non-linear trajectory instead of the piecewise linear trajectory.

Smoothed piecewise constant In the HDMs proposed by Richards & Bridle (1999) and Gao et al. (2000), the piecewise constant target sequence is smoothed using a simple Kalman smoother (c and ρ denote the target and time-constant). Each state $x(t)$ at time t along the smoothed state sequence can be seen as a weighted sum of the target at that time and the adjacent values: $x(t-1)$ and $x(t+1)$. The Kalman smoother equations which describe the smoothed trajectory were presented in (Richards & Bridle 1999).

Causal low-pass filter Deng (1998) propose a statistical HDM in which the VRT dynamics is described by a causal second-order discrete-time critically-damped unity-gain low-pass filter. The state equation using the state-space formalism is given in Table 2.2, where ρ determines the rate of the critically-damped system dynamics towards the local target c. In addition, the global smoothness was achieved by imposing continuity constraints in such a way that the state vector $x(t)$ at the end of a dynamic regime (e.g., phone) to be identical to the initial state vector for the immediate following regime. Therefore, the model is able to characterise long-span coarticulation effects. A simplified first-order state equation was used in (Deng & Ma 2000). In the 1998 Workshop on Language Engineering (Bridle et al. 1999), Deng's statistical HDM provided similar level of performance compared to the HDM proposed by Richards & Bridle (1999).

Hidden-trajectory HMM Zhou et al. (2003) proposed a hidden-trajectory HMM (HTHMM) which combines the advantages of both HDMs and HMMs. The hidden trajectory model was based on the same state equation as in (Deng 1998), though the frame-wise noise term is removed to allow simple training and recognition implementations. This hidden trajectory model was incorporate into a standard HMM whose Gaussian means in the output mixture distributions were adapted to the hidden trajectory model. A full MAP decoding algorithm for HTHMMs was derived and described in (Seide, Zhou & Deng 2003), which is the first recognition system in the context of HDMs. Using a small vocabulary digit-string corpus, the context-independent HTHMMs achieved slightly better performance compared to triphone HMMs with about 20% few parameters.

2.4.4 Articulatory-to-acoustic mapping

In the HDM framework, an important component is the articulatory-to-acoustic mapping which transforms the hidden dynamics to the acoustic domain. Both linear and non-linear mappings have been used in the past, and in either case, the mapping parameters can be tied across different phones based on different categorisations of the phone set.

Non-linear mapping

In most HDM approaches, a non-linear mapping such as an MLP is used. This is natural since it is widely established that the relationship between the hidden representation, e.g., articulator movement or formant frequencies, and the acoustic representation is non-linear (Richards & Bridle 1999, Russell et al. 2007). For example, a single MLP for all phone classes was used by Richards & Bridle (1999), while an ensemble of 10 MLPs was used in (Deng & Ma 2000), with each MLP associated with a distinct manner of articulation. The ten phone classes are listed in Table 2.3 below. As can be seen, all vowels are tied using a single MLP since vowel distinction is based exclusively on different target values in the intermediate VTR domain. For phones with similar VTR targets, e.g., [s] and [sh], separate MLPs are used. ‘Sil’ and ‘sp’ are two silence models so that they can be tied using a single mapping.

Number	Phone class
1	aw, ay, ey, ow, oy, aa, ae, ah, ao, ax, ih, iy, uh, uw, er, eh, el
2	l, w, r, y
3	f, th, sh
4	s, ch
5	v, dh, zh
6	z, jh
7	p, t, k
8	b, d, g
9	m, n, ng, en
10	sil, sp

Table 2.3: The phone categorisation scheme used by Deng & Ma (2000) for the VTR-to-MFCC (articulatory-to-acoustic) mapping based on the TIMIT phonetic labelling system.

Linear mapping

Examples using linear articulatory-to-acoustic mappings include (Zhou et al. 2003, Seide et al. 2003, Ma & Deng 2004, Russell & Jackson 2005). In (Zhou et al. 2003, Seide et al. 2003, Ma & Deng 2004), a mixture of linear mappings was used to approximate the non-linear mapping (MLP) previously described in (Deng 1998, Deng & Ma 2000). The same mixture component is used for all frames of a speech unit in (Ma & Deng 2004), while in (Zhou et al. 2003, Seide et al. 2003) the mixture component is chosen independently in each frame. The linear mapping used in (Russell & Jackson 2005, Russell et al. 2007) is estimated based on matched formant and acoustic data before the estimation of any other model parameters takes place (see Section 4.4 for more details).

Although linear articulatory-to-acoustic mappings do offer some computational advantages (especially during the model training stage), they are clearly less realistic than their non-linear counterparts. In (Russell et al. 2007), for example, the authors found that their linear formant-to-acoustic mappings rely heavily on the band-energy type of data, which is linearly related to the acoustic representation, and are not actually using the formant frequency information. This result exposes the limitations of linear mappings and provides empirical evidence of the need for non-linear formant-to-acoustic mappings.

2.5 Summary

This chapter reviews a variety of approaches which attempt to characterise speech dynamics. Section 2.3 discusses the family of SHMMs and Section 2.4 outlines various HDMs. SHMMs provide an improved model of feature dynamics at the segment level, relative to a frame-based HMM, in the surface, acoustic representation of speech. Trajectory-based SHMMs model dynamics by means of some form of trajectory in the acoustic space. Many different types of trajectory have been reviewed, including constant, linear, higher order polynomial and non-parametric. A linear dynamical system can be used as a general acoustic model, which includes many of the other acoustic models as special cases.

HDMs have theoretical advantages to model coarticulation and long-span dependencies over the entire utterance in an articulatory-related intermediate layer. An HDM can be interpreted from the perspective of a state-space model, in which the state equation describes the smooth motion of the articulators, which is projected to the acoustic layer through an articulatory-to-acoustic mapping. Most HDM approaches use linear target filtering as a model for the hidden dynamics and artificial neural networks as the non-linear mapping function in the observation equation.

Chapter 3

Speech data

The purpose of this chapter is to provide a detailed description of the speech data used in this thesis work. Two types of data have been used: acoustic data and formant data, and these are described in Sections 3.1 and 3.2, respectively. Section 3.3 then gives a short description of the speech corpora used in this research.

3.1 Acoustic data

In current ASR systems, feature extraction is an important component because the outcome of speech signal analysis is the foundation of the subsequent acoustic modelling and testing. The aim of front-end processing is to transform the speech waveform into a sequence of acoustic feature vectors. The ideal acoustic features would:

- reflect the human speech production mechanism - e.g., the source-filter model.
- reflect knowledge of the human peripheral auditory system, e.g., the non-linearity of the human perceptual frequency scale.
- maintain all perceptually important information while suppressing other unimportant information (e.g., phase or noise).
- be computationally efficient to allow real-time applications.

As in previous study on MSHMMs (Russell & Jackson 2005, Russell et al. 2007) and many other ASR systems, mel-frequency cepstral coefficients are chosen as the acoustic features in this research. In this work, the calculation of MFCCs is performed using standard tools in HTK, as described in the following paragraphs.

The speech waveform is first divided into equally spaced (10ms spacing) and overlapping short segments of fixed duration of 25ms, based on the underlying assumption that the shape of the vocal tract (and hence the speech spectral characteristics) are approximately stationary over an interval of 20-30ms. Assume that each segment contains N speech samples $\{s_n : s_1, s_2, \ldots, s_N\}$. A Hamming window function is then applied to taper the samples in each segment so that discontinuities at the segment edges are attenuated (Young et al. 2005). The Hamming window function is given by

$$s_n^{'} = \left\{0.54 - 0.46\cos\left(\frac{2\pi(n-1)}{N-1}\right)\right\}s_n. \tag{3.1}$$

Next, a Discrete Fourier Transform (DFT) is applied to the windowed speech waveform, resulting in a $N/2$-point complex spectrum. The modulus is taken to produce the power spectrum, with the effect of removing the phase information from the acoustic representation, which is of little perceptually importance. Then a logarithm is applied to the power spectrum to generate the log-power spectrum, sometimes referred to as the short-term spectrum. The use of a logarithm is motivated by results from psycho-acoustics which show that loudness is perceived on a logarithmic scale (see, e.g., Moore, Glasberg & Vickers (1999)). In addition, the use of the logarithm has the effect to compress the dynamic range.

So far, each speech segment is represented by a $N/2$-point log-power spectrum, which is distributed linearly on the frequency scale. However, psycho-acoustic experiments show that the perceptual frequency scale is not linear but non-linear (see, e.g., Zwicker & Fastl (1999)). One approximation to this non-linearity of perceptual scale is the so-called mel-scale, which is approximately linear up to about 1,000Hz and logarithmic above 1,000Hz. A popular approach to converting a value f in Hertz into mels is given by O'Shaughnessy (1987) as:

$$\text{Mel}(f) = 2595\log_{10}(1 + \frac{f}{700}). \tag{3.2}$$

A mel-scale triangular filter bank (Young et al. 2005) is then used to smooth the log-power spectral estimates to produce a mel-scale log-power spectrum. A finite number M of triangular filters are chosen and arranged equally spaced along the mel-scale, which, given the correct bandwidth ($\approx$ 100 mels), translate into critical band filters on the linear frequency scale. The mel-scale filter bank produces an M-point mel-scale log-power spectrum, in which each point $\{m_j\}$ ($1 \leq j \leq M$) is a weighted sum of the log-power spectral coefficients, representing the spectral magnitude in the corresponding filter bank channel j.

Finally, a discrete Cosine transform (DCT) is applied to the mel-scale log-power spectrum, resulting in a set of M mel frequency cepstral coefficients $\{c_1, \ldots, c_M\}$, where

$$c_i = \sqrt{\frac{2}{M}} \sum_{j=1}^{M} m_j \cos\Big(\frac{\pi i}{M}(j - 0.5)\Big) \tag{3.3}$$

and $\{m_j\}$ ($1 \leq j \leq M$) denote the filter bank amplitudes. The DCT has the effect of compressing the spectral information into the lower order coefficients and it also decorrelates them allowing the subsequent statistical modelling to use diagonal covariance matrices (Young 1995). In this research, the first 12 MFCCs $\{c_1, \ldots, c_{12}\}$ plus c_0 (which is proportional to the average log power spectrum component value) are used as the basic acoustic feature vector. Dynamic features (Δ and Δ^2 parameters) are also used in some experiments in this study, which are calculated based on Equation (2.2).

3.2 Formant data

3.2.1 Formants in ASR

It has been known for many years that formant frequencies are important in determining the phonetic content of speech sounds (Holmes & Garner 2000). For example, the first three formant frequencies are sufficient to discriminate all English vowels and semivowels (Peterson & Barney 1952). In fact, experts in spectrogram reading (Zue 1991) can often crack the ‘code’ behind the speech signal by observing the formant structure in the speech spectrogram alone with a high degree of reliability. Moreover, formant frequencies are of primary importance for speech perception. It is found that an overall 10% change in formant frequencies makes a dramatic change, though the change is primarily in one's perception of the characteristics of the speaker rather than of the phonetic content. For example, a 30% increase in the frequency of F_1 results in the sentence originally spoken in a British accent to be perceived by some to be spoken by an Australian. Formant frequencies above F_3 and formant bandwidths, though, have little perceptual effect (Hunt 1987).

Formant dynamics are closely correlated with dynamics of the principle articulators, such as tongue, lips and jaw. Generally speaking, the lower the value of F_1 is, the closer the tongue is to the roof of the mouth. On the other hand, the F_2 value is proportional to the frontness or backness of the highest part of the tongue during the production of the vowel. Therefore, formant trajectories in the speech spectrogram (again see Figure 1.2 on page 9) tell us about, roughly, how the shape of the vocal tract changes with time.

Despite the phonetic significance of formants, formant frequencies or/and amplitudes alone as feature vectors in ASR is rare. This is partly because that automatic estimation of formant frequencies is difficult and prone to error. While formants are reasonably well defined for voiced sounds, a formant for unvoiced sounds (such as some stops and fricatives) may be so weak as a consequence of weak excitation that it causes no peak in the spectrum. Formant estimation for nasal sounds poses another difficulty, since the basic speech production theory does not strictly apply to this type of speech sounds because of the concurrent spectral zeros. In addition, the chosen formant representation may not contain sufficient information on its own for accurate speech recognition (Russell & Jackson 2005).

In practice, formant information is usually used as a supplement to acoustic features to aid speech recognition. Several authors have confirmed that an appropriate combination of formant features and acoustic features could provide some recognition performance advantages over the traditional, cepstrum features only approach (Holmes, Holmes & Garner 1997, Garner & Holmes 1998, Holmes & Garner 2000, Wilkinson & Russell 2002). Other researchers took advantage of the formant information in an intermediate space to explicitly account for dynamics. Examples include (Deng 1998, Deng & Ma 2000, Zhou et al. 2003, Russell & Jackson 2005, Russell et al. 2007), as described in hidden dynamic models in Section 2.4.3. Indeed, the fact that formants are closely related to both acoustics and articulation makes them a good choice for the intermediate representation in HDMs.

The particular type of formant data employed in this work is the 12-dimensional PFS control parameters, which belongs to one of three alternative formant-based intermediate representations studied previously in a linear/linear MSHMM. The use of this special type of formant-based representation sets this research apart from others because the 12 PFS control parameters are originally from a formant synthesiser. This is actually a more immediate motivation for the development of an MSHMM as a unified model for speech recognition and synthesis. The rest of this section first describes what the 12 PFS parameters are (Section 3.2.2), and then shows how these parameters are calculated (Section 3.2.3) and how they are initially used in a formant synthesiser for speech synthesis (Section 3.2.4).

3.2.2 The 12 PFS control parameters

Number	Parameter description
1	F_N: Frequency of ‘low frequency’ formant (default value 250Hz)
2	A_{LF}: Amplitude of F_N in dB
3	F_1: Frequency of first formant (in 25 Hz steps)
4	A_1: Amplitude of first formant in dB
5	F_2: Frequency of second formant (in 50 Hz steps)
6	A_2: Amplitude of second formant in dB
7	F_3: Frequency of third formant (in 50 Hz steps)
8	A_3: Amplitude of third formant in dB
9	A_{HF}: Amplitude in high frequency region (centred on 3,500 Hz) in dB
10	V: Degree of voicing (1=completely unvoiced, 63 = fully voiced)
11	F_0: Fundamental frequency on logarithmic scale
12	MS: Glottal-pulse mark-space ratio

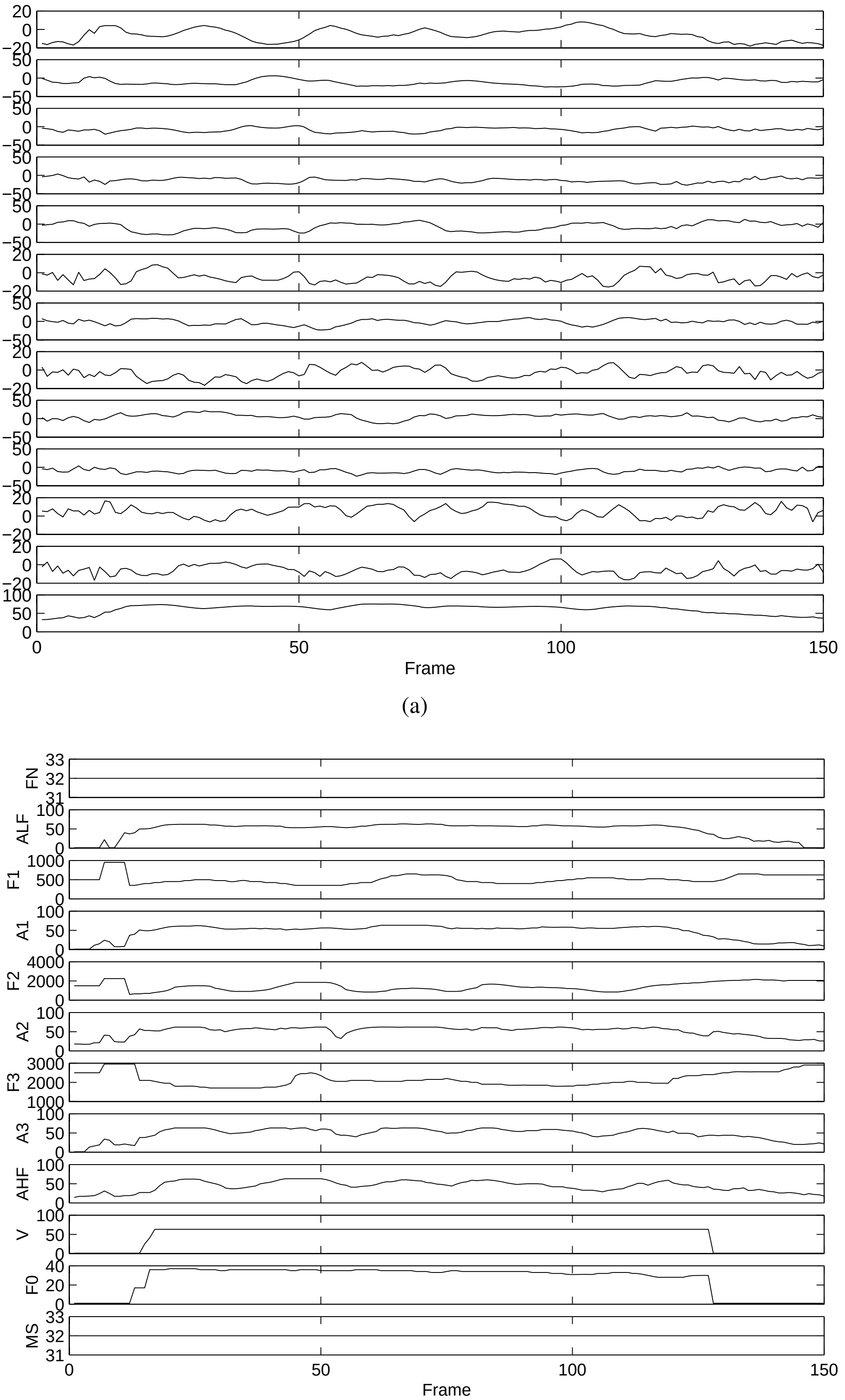
Table 3.1: The 12 parallel formant synthesiser control parameters. Taken from (Russell, Zheng & Jackson 2007).

The 12 PFS control parameters are summarised in Table 3.1. Among the 12 parameters, the first 9 parameters reflect the spectral shape of the vocal tract, while the remaining 3 parameters define the sound source. These 12 parameters are generated from the Holmes formant analysis toolkit (Holmes 1998, Holmes 2001), which is described in Section 3.2.3. For the first and twelfth parameters (i.e., F_N and MS in Table 3.1), the Holmes formant analyser currently returns fixed values. The tenth and eleventh parameters, i.e., degree of voicing V and fundamental frequency F_0, are obtained based on multi-channel autocorrelation analysis (Holmes 1998), which will be described in more detail in Section 3.2.4.

Figure 3.1 compares the 13 MFCCs and 12 PFS parameters of an example TIMIT sentence "Where were you while we were away?" (sx9), spoken by speaker msjs1. Recall that in the wide-band speech spectrum of the same utterance (see Figure 1.2 on page 9), we observed that formant trajectories, especially the formant trajectories for F_1, F_2 and F_3, are mostly smoothly time varying due to the constraints of the human vocal tract. The 12 PFS control parameters, as shown in Figure 3.1 (b), demonstrate this smoothness again, as most of the 12 PFS parameters vary slowly and smoothly. On the other hand, the corresponding 13 MFCCs, especially the higher order coefficients, may often vary rapidly and erratically, as can be seen in Figure 3.1 (a). The comparison between these two types of data in Figure 3.1 shows the advantage of modelling dynamics directly in a formant-based domain. Many phenomena which may be characterised simply in terms of formant transitions (or formant trajectories) are often exhibited in the acoustic domain as complicated paths which span several different frequency bands. This is a principal motivation for the incorporation of such a formant-based representation in our MSHMMs.

3.2.3 Formant analysis

There are a number of techniques to automatically estimate formant frequencies from acoustic speech signals. The conventional method of estimating formant frequencies is based on the standard LPC technique (Markel & Gray 1976). The underlying assumption of the LPC formalism is that the human vocal tract can be modelled as a linear filter with a small number of poles but no zeros in its transfer function. Apparently, this assumption is untrue for nasal sounds due to the introduction of zeros caused by the inclusion of the nasal cavity



(a)

(b)

Figure 3.1: (a) The 13 MFCCs $\{c_1, c_2, \ldots, c_{12}, c_0\}$, displayed from top to bottom. (b) The 12 PFS control parameters. Features derived from the TIMIT utterance “Where were you while we were away?” (`sx9`), spoken by speaker `msjs1`.

into the vocal tract. This is one of the objections to LPC for feature extraction in speech recognition. As far as this research is concerned, the J. N. Holmes formant analyser (Holmes 2001) is used to extract the formant information, which is described in detail in this section.

The Holmes formant analyser

The Holmes formant analysis toolkit (Holmes 1998, Holmes 2001) automatically transforms a speech waveform into three alternative formant-based parameterisations. All these three representations have been used in the previous experimental work on a linear/linear MSHMM, which are referred to as 3FF, 3FF+5BE and 12PFS, as described in Section 4.5.

The following description of the calculation of the 12 PFS control parameters largely follows that of Holmes (2001). An important feature of the Holmes formant analyser is that the direct measurement of formant frequencies is replaced with a *matching process*. Figure 3.2 shows the main steps in formant estimation using the Holmes formant analyser.

The speech signal is sampled at 8,000 samples per second, which gives a bandwidth of 4KHz. Pitch-synchronous Fourier analysis is applied to generate a 31-dimensional log power spectral vector for every 10ms of input speech, representing frequencies from 125 Hz to 3875 Hz. At the heart of the Holmes formant analyser is a ‘codebook’, which contains one hundred and fifty reference spectra. These reference patterns represent the range of sets of formant frequencies which occur in all possible speech sounds. The number of reference spectral cross-sections may vary, though it is found that one hundred and fifty is adequate (Holmes 2001). Each reference spectrum in the codebook is associated and stored with one or possibly more sets of the first three formant frequencies, which have been determined by a human expert from examination of the corresponding spectral cross-section.[1]

For each input spectrum, a shortlist of reference spectra from the codebook is selected based on the comparison in terms of general aspects of their shape. In (Holmes & Holmes 2001), this is achieved by selecting six candidates from the codebook which have the lowest squared Euclidean distances with the input spectral cross-section. Ideally, every entry in

[1] The use of alternative sets of formant frequencies for a reference spectrum happens when it is impossible to be certain which spectral peak should be associated with each of the formants.

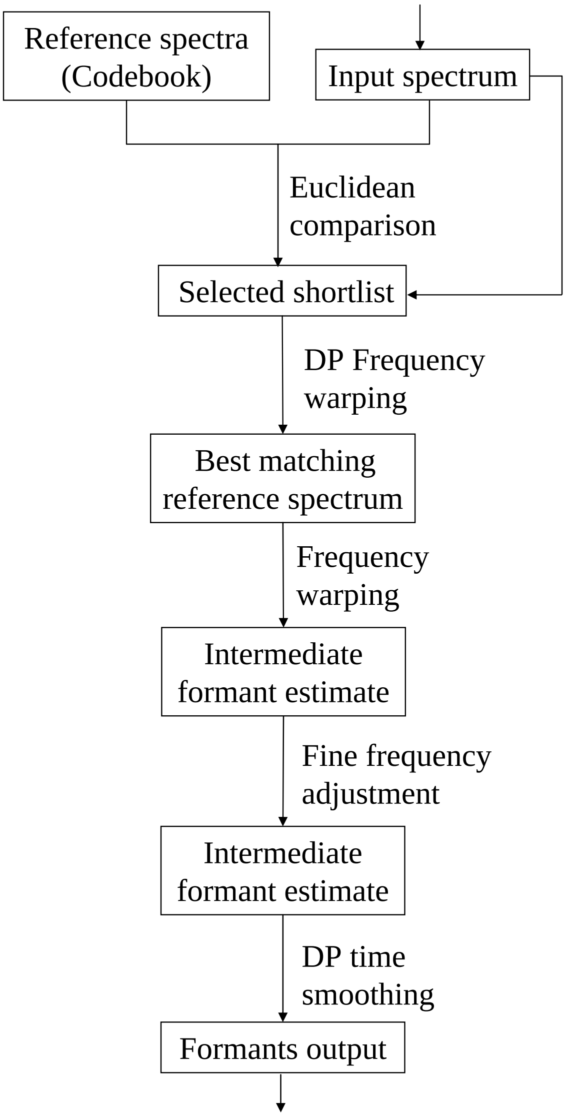

Reference spectra (Codebook)
Input spectrum
Euclidean comparison
Selected shortlist
DP Frequency warping
Best matching reference spectrum
Frequency warping
Intermediate formant estimate
Fine frequency adjustment
Intermediate formant estimate
DP time smoothing
Formants output

Figure 3.2: Figure showing the main steps of estimating formant frequencies using the Holmes formant analyser.

the codebook should be considered throughout the whole formant analysis process until the formant frequencies for the unknown spectrum are assigned. However, using a sub-set of the codebook (i.e., the shortlist) for further analysis can significantly reduce the computation associated with the matching process.

Once the shortlist is chosen, a more detailed comparison is made between the input spectra and each reference spectrum in the shortlist. This involves the use of dynamic programming (DP) (Bellman 2003) to undertake constrained, non-linear frequency warping. As a result, the best matching reference spectrum in the shortlist is found, which gives the minimum DP distance with the input spectrum. The formant frequencies associated with this best matching reference pattern are adjusted according to the frequency warping and the results are used as formant frequency estimates for the input cross-section.

The formant frequencies are further refined to take account of the shape of the input spectrum near to the chosen formants to obtain more accurate formant estimates. For example, parabolic interpolation is employed to find a frequency between the two spectral points either side of the determined formant frequency, such that the frequency so found is at the highest point of a parabola which passes through the spectral cross-section at the determined formant frequency and at its two neighbouring frequencies (Holmes 2001).

Finally, DP time smoothing is used to choose a unique set of formant frequencies at each time t, taking into consideration the continuity constraints of the formant trajectories prior to and subsequent to time t. This is motivated by the fact that for each reference spectrum in the codebook, there are possibly more than one set of stored formant frequencies. Sometimes the best matching reference spectrum for a particular input frame may give only one unique set of formant frequencies, while a neighbouring frame may give alternative sets of formant frequencies. Moreover, in the case that multiple best reference spectra are chosen from the selected shortlist, there will be a number of sets of formant estimates to choose from. The DP based time smoothing therefore selects between alternative sets of formant frequencies derived from all members of the shortlist to produce formant trajectories that have the minimum discontinuity in time.

Given the formant frequencies obtained from the DP time smoothing, the corresponding formant amplitudes are determined from input cross-section amplitudes at estimated formant frequencies. The output of the formant frequencies and amplitude are generated every 10ms.

3.2.4 Formant synthesis

Formant synthesis is based on a formant level description of the vocal tract transfer function. Unlike the conventional waveform concatenation synthesis, which generates speech by concatenating stored examples of actual speech from a database (Hunt & Black 1996), no corpus of recorded human speech samples is needed at runtime in formant synthesis. A number of formant synthesisers have been developed in the past (Holmes et al. 1964, Hunnicutt & Klatt 1987). This section describes a formant synthesis system based on the classic HMS formant synthesiser (Holmes et al. 1964), developed at the Joint Speech Research Unit (JSRU), in the UK. The HMS formant synthesiser is also used in this research to generate speech, as described in Chapter 6.

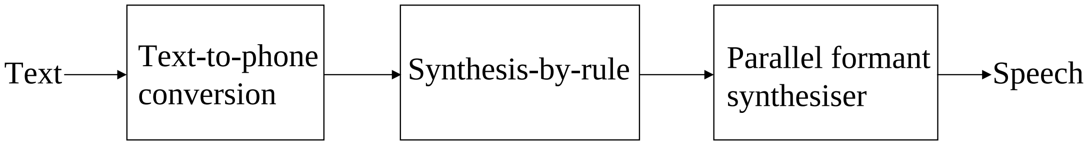


Figure 3.3: Block diagram showing the main components of the JSRU synthesis-by-rule formant synthesis system.

The JSRU formant synthesis system is a rule-based synthesiser which comprises three main components, as shown in Figure 3.3. Firstly, a rule-based text-to-speech component converts orthographic text into a sequence of phonetic element, each of them corresponding to an entry in the talker table[2]. These rules are often referred to as pronunciation rules or letter-to-sound rules. It should be noted that phonetic elements and phonemes are not equivalent here. Some phonemes are represented by two vowel-like elements, for example, diphthongs. Other phonemes, such as stops, are treated as a sequence of three elements, corresponding to the period of closure, the period of high energy at the beginning of the release, and the period of diminished energy at the end of the release (Holmes et al. 1964).

Secondly, the role of the synthesis-by-rule component is to generate a sequence of synthesiser control parameters from a stream of phonetic elements obtained in the previous stage. A talker table is the key information source at this stage, which contains parameters that specify each phonetic element. For each phonetic element the talker table contains the duration and target frequencies and amplitudes of the first three formants which characterise

[2] The talker table is described in the following paragraph. In the original work (Holmes et al. 1964), there are totally 50 phonetic elements.

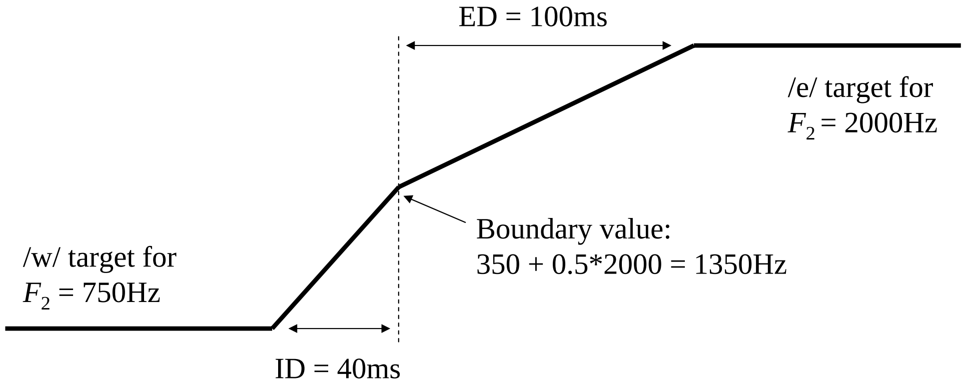


Figure 3.4: Second formant transition for the phonetic element sequence [we] according to the Holmes-Mattingly-Shearme synthesis-by-rule system (from Holmes & Holmes (2001)).

that speaker's target instantiation of the sound during its steady-state period. Moreover, a set of rules governing transitions to and from these target values are also specified and stored in the table, such as the internal duration (ID) and external duration (ED), the fixed contribution to the boundary value and the proportion of the target frequency which contributes to its boundary value.

For each input phonetic element, a sequence of sets of synthesiser control parameters must be calculated in order to synthesise that sound. The number of such sets of parameters is equal to the duration of the element. The synthesiser control parameter values are determined using information contained in the talker table for the current phonetic element and its immediately preceding and following elements in the input sentence. For example, linear interpolation rules are applied to ensure smooth formant trajectories are generated, which is achieved by calculating a boundary value during the transition from one target to another.

Figure 3.4 shows an example of the interpolation rules applied to the second formant transition for the phonetic sequence [we] as in the word well. The talker table for the phonetic element [w] has the following parameters for the formant frequency F_2:

- The target frequency - 750Hz
- Four other parameters which control the transition out of the phone [w]
  - The fixed contribution to the boundary value - 350Hz

- The proportion of the target frequency (steady-state value) of the adjacent element which contributes to its boundary value - 0.5
- The internal duration (i.e., the duration of the transition before the boundary) - 40ms
- The external duration (i.e., the duration of the transition after the boundary) - 100ms

Since the target value for F_2 of the phonetic segment [e] is 2,000Hz, the boundary value F_2^b of F_2 in the transition from [w] to [e] is given by the fixed contribution from [w] plus the given proportion of the following target frequency, i.e., $F_2^b = 350 + 0.5 \times 2000 = 1350$.

Finally, the parallel formant synthesiser described in (Holmes et al. 1964), as shown in Figure 3.5, is used to synthesize speech given a sequence of synthesiser control parameters. This formant synthesiser is also used in the research presented in this thesis to generate synthetic speech. The parallel formant synthesiser consists of a series of three resonators, arranged in parallel, which correspond to the first three formants of the vocal tract. Energy of higher frequency is represented by a fourth resonator. The formant resonators are configured in parallel to enable separate mixing of the voiced and unvoiced excitation for each formant, with potential advantages in generating high quality synthetic speech. A sequence of 12 PFS control parameters is sent to the parallel formant synthesiser every 10 ms to generate speech. (The 12 PFS control parameters are listed in Table 3.1 on page 56.)

For formant synthesis, it is very important to get the excitation right. The source of the excitation for the filters is determined by a voicing control. The voicing control is given in the talker table for the current phonetic element and remains the same for the duration of that element. For voiced excitation a waveform with a particular fundamental frequency F_0 is used, whereas for unvoiced excitation a random noise source is used. In the original work (Holmes et al. 1964), the series of fundamental frequency values is derived by measuring a spectrogram of a human version of the utterance to be synthesized, or is simply estimated. The fundamental frequency F_0 is normally specified in the input sentence once for each elements, and is subject to linear interpolation at the boundaries between adjacent elements.

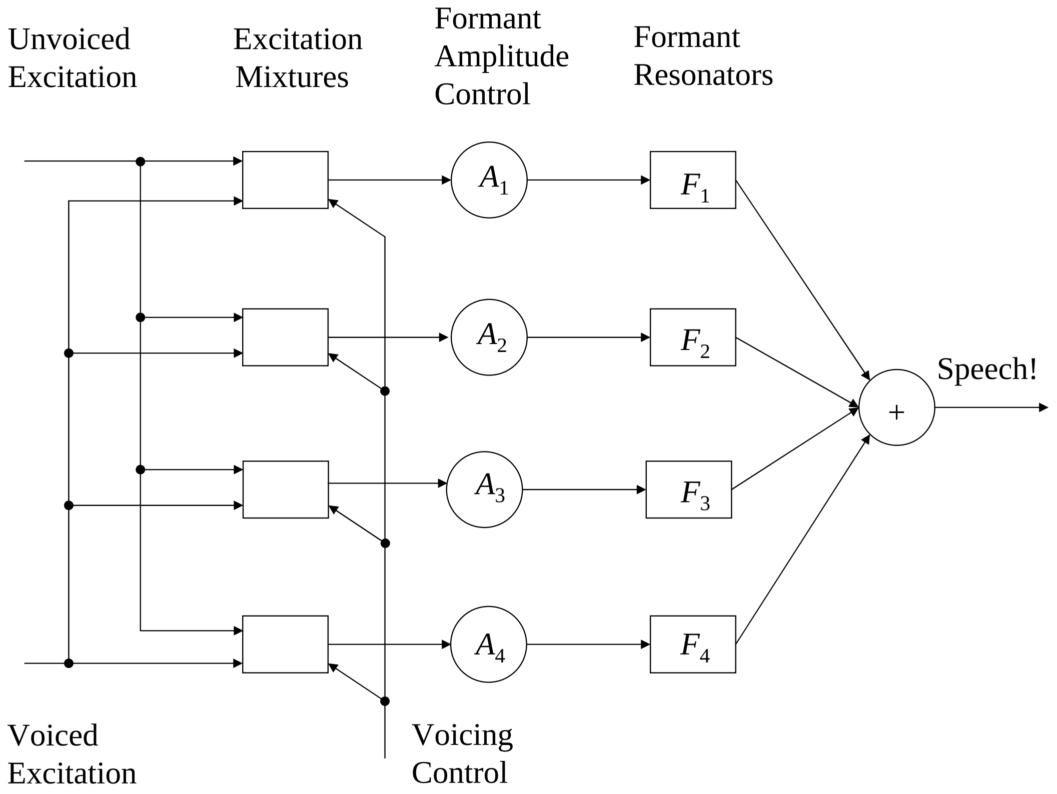


Figure 3.5: Schematic diagram of a parallel formant synthesiser. Reproduction of the figure in (Russell 2002).

Later work reported in (Holmes 1998) described a method to estimate fundamental frequency F_0 based on multi-channel autocorrelation analysis, as explained below.

Short-time autocorrelation is a well-established method for estimation of the F_0 because of the signal's tendency to resemble itself with a delay corresponding to the fundamental frequency. In (Holmes 1998), the speech is filtered into eight separate frequency bands, representing the lowest 500 Hz and seven overlapping band-pass channels each about 1000 Hz wide. The outputs of all the band-pass channels are full-wave rectified and band-pass filtered between 50 Hz and 500 Hz. Autocorrelation functions are calculated for the signals from all eight channels, and these functions are combined with appropriate weights. The highest peak (when several consecutive peaks are of about the same height, the earliest peak is the required one) in the combined autocorrelation function is at the fundamental period.

Copy synthesis, where the synthesiser control parameters are derived directly from a natural utterance and then used to re-synthesise that speech, has shown that given a appropriate set of synthesiser control parameters, a parallel formant synthesiser can generate extremely high quality speech which is almost indistinguishable from the natural (Holmes 1973). The implication of this result is that the robotic synthetic speech quality, which is typically associated with this type of synthesiser, is because of the errors introduced by the processes which convert a string of text into a sequence of synthesiser control parameters, rather than shortcomings of the parallel formant synthesiser (Russell 2003). Hence, the major challenge in formant synthesis is to obtain appropriate synthesiser control parameters. In addition, Holmes (1989) shows that, at least for female voices, an additional higher frequency formant leads to a significant improvement in the quality of the synthetic speech.

3.3 The TIMIT speech corpus

All experimental work presented in this thesis is based on TIMIT. The TIMIT speech corpus is widely used in ASR research. The corpus contains read speech of 630 speakers (438 males and 192 females), divided into 8 dialect regions, which represent 8 different dialect areas of the United States. Ten phonetically-balanced sentences, including 2 fixed dialect sentences (SA sentences) which are designed to reflect dialectal variants of the speakers, are read by all 630 speakers.

Not all data in TIMIT were used in previous work on MSHMMs. For example, the MSHMM proposed in (Russell & Jackson 2005) was trained and therefore tested on data from the male subjects only of the TIMIT corpus. Although later work on the MSHMM (Russell et al. 2007) used data from female speakers of TIMIT, this data was only used to build gender-dependent models for female speakers. The reason for using gender-dependent (rather than gender-independent) models in the MSHMM framework is to reduce the variability that the experimental MSHMM system would need to accommodate. As we know, one simple approach to overcoming the variability problem in speech pattern modelling is to remove it. For example, speaker-dependent systems are often used to avoid inter-speaker variability. Given the TIMIT database, using gender-dependent data such as male data can remove the inter-gender variability so that the resulting model may be more accurate in characterizing the underlying mechanisms for the current task domain. Another reason for using the data of male speakers only is that initially there was a concern that formant analysis (since the intermediate layer of MSHMMs is based on formant frequencies) for female speakers is more difficult than for male speakers because of their relatively higher pitch (Holmes 2001), and therefore may introduce errors and other sources of variability which could deteriorate the model. However, Russell et al. (2007) show that there is no evidence that performance for female speech is affected by difficulties with formant analysis.

On the other side, a major goal of the MSHMM is to seek a more efficient means of modelling coarticulation, without the need for a large amount of training data. In doing so, the MSHMM includes a formant-based intermediate layer where dynamics can be modelled simply, and makes a couple of simple assumptions, such as a fixed, linear trajectory in the intermediate layer to model dynamics and a single Gaussian PDF in the acoustic domain to account for acoustic variability (see Section 4.2). However, these simplifying assumptions are less realistic and may lead to a model which is not sufficiently flexible to accurately deal with speech data which contain a wide range of variability sources. Therefore, focusing on a slightly constrained task domain (such as the gender-dependent male system as described in (Russell & Jackson 2005)) is a valid strategy for the development of this type of MSHMMs, especially at the early stage of this model.

The focus of this research is still on the MSHMMs for male speakers, though female data are also used. In particular, recognition results are presented for MSHMMs which are built on the full TIMIT training set (including both male and female speakers). This is the first time in the context of the MSHMM that gender-independent models are used. It would be interesting to see how the increased amount of training data and the added degree of variability (introduced by the gender) affect the performance of the MSHMM.

3.4 Summary

This section describes in detail the speech data used in this study. As in previous research on MSHMMs, MFCCs are employed as acoustic features, and all the 12 PFS control parameters are used as formant features. These two types of data are calculated by using the HTK and the Holmes formant analyser (see Section 3.2.3), respectively. The TIMIT corpus of read speech is used for all experiments presented in this work.

The use of 12 PFS control parameters (see Table 3.1) is a distinguishing feature of this research. These parameters are originally used for formant synthesis, as described in Section 3.2.4. Formant synthesis is a model-based approach to speech synthesis, which builds on a formant level description of the vocal-tract transfer function. Phonetic units are handcrafted and parameters (such as formant frequencies and amplitudes) are stored in a talker table, which can be updated as critical listening suggests. The formant synthesiser control parameters are derived by rules, in conjunction with data from the talker table. Linear interpolation rules are typically used to ensure that smooth formant trajectories are generated.

Chapter 4

Multiple-level segmental HMMs

4.1 Introduction

This chapter presents a detailed description of the framework chosen for this research: the multiple-level segmental HMM. The key features of a general multiple-level segmental HMM include:

- a segmental HMM framework, which offers
  - an improved model of dynamics in which the relationship between speech frames within a segment can be modelled explicitly in terms of a trajectory model
  - a tractable mathematical framework with straightforward training and recognition algorithms
- an intermediate layer, which provides an articulatory-related space where dynamics and coarticulation can be dealt with directly and naturally
- an articulatory-to-acoustic mapping, which transforms the articulator dynamics or formant transistions at the intermediate layer into the acoustic realisation of speech.

Specifically, this chapter covers all theoretical aspects of a linear/linear MSHMM (Russell & Jackson 2005, Russell et al. 2007), the first implementation of an MSHMM, including the formulation of the model (Section 4.2), the segmental Viterbi decoder (Section 4.3) and the model parameter estimation algorithm (Section 4.4). Finally, Section 4.5 provides some of the previous experimental results and conclusions.

4.2 Formulation of the linear/linear MSHMM

From the perspective of mathematics, a linear/linear MSHMM (Russell & Jackson 2005) is derived from a linear FTSHMM (Gish & Ng 1993, Deng et al. 1994, Holmes & Russell 1999, see Section 2.3.4 on page 30) by incorporating an intermediate 'articulatory' layer. A linear/linear MSHMM $\mathcal{M}$ has a finite number of N states $\mathcal{S} = \{s_1, \ldots, s_N\}$. Each state $s_j \in \mathcal{S}$ $(1 \leq j \leq N)$ is associated with a midpoint vector $\boldsymbol{c}_j$ and a slope vector $\boldsymbol{m}_j$ in the D-dimensional intermediate space. The midpoint and slope vectors have the same dimension as that of the intermediate layer. These two parameters specify a linear trajectory in the intermediate space. For example, a τ-length segment trajectory is given by

$$\boldsymbol{f}_j(t) = (t - \bar{t})\boldsymbol{m}_j + \boldsymbol{c}_j, \tag{4.1}$$

where $\bar{t} = (\tau + 1)/2$ is the midpoint of the start and end time points of the segment. In addition, the MSHMM $\mathcal{M}$ is also associated with one of K phone-class-dependent, linear, articulatory-to-acoustic mappings, $\boldsymbol{W}_k$, where $k \in \{1, \ldots, K\}$. It should be noted that the mapping $\boldsymbol{W}_k$ is model-dependent, rather than state-dependent, which is 'shared' by all states in the MSHMM $\mathcal{M}$.

The linear trajectory $\boldsymbol{f}_j$ in the intermediate layer is projected onto the acoustic space via the linear articulatory-to-acoustic mapping $\boldsymbol{W}_k$. The transformed trajectory $\boldsymbol{W}_k(\boldsymbol{f}_j)$ in the acoustic layer is also linear because of the linear nature of both the trajectory $\boldsymbol{f}_j$ and the mapping $\boldsymbol{W}_k$. Comparison is then made frame by frame between the unknown acoustic observations and the transformed trajectory $\boldsymbol{W}_k(\boldsymbol{f}_j)$. The probability of the acoustic observation sequence $\boldsymbol{o}_1^\tau = \{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_\tau\}$ given state s_j is (ignoring any duration probability)

$$b_j(\boldsymbol{o}_1^\tau) = \prod_{t=1}^{\tau} \mathcal{N}\Big(\boldsymbol{o}_t | \boldsymbol{W}_k\big(\boldsymbol{f}_j(t)\big), \Sigma_j\Big), \tag{4.2}$$

where $\mathcal{N}\Big(\boldsymbol{o}_t | \boldsymbol{W}_k\big(\boldsymbol{f}_j(t)\big), \Sigma_j\Big)$ denotes a Gaussian PDF with mean $\boldsymbol{W}_k\big(\boldsymbol{f}_j(t)\big)$ and covariance matrix $\boldsymbol{\Sigma}_j$, evaluated at $\boldsymbol{o}_t$.

Suppose that an observation sequence $\boldsymbol{O} = \{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_T\}$ comprises several segments. Now consider the problem of computing the probability $p(\boldsymbol{O}|\mathcal{M})$ of $\boldsymbol{O}$ generated by the MSHMM $\mathcal{M}$. To simplify notation, the underlying state process is assumed to be a left-to-right model. Given the MSHMM $\mathcal{M}$, the observation sequence $\boldsymbol{O}$ is generated via a state sequence (or segmentation) $l_1^N = \{l_1 \times s_1, \ldots, l_N \times s_N\}$, where $l_i \times s_i$ denotes a duration l_i spent in state s_i for each $i \in \{1, \ldots, N\}$, and $l_1 + l_2 + \ldots + l_N = T$. Therefore, the probability of $\boldsymbol{O}$ given $\mathcal{M}$ is

$$p(\boldsymbol{O}|\mathcal{M}) = \sum_{l_1^N} p(\boldsymbol{O}, l_1^N|\mathcal{M}) = \sum_{l_1^N} p(\boldsymbol{O}|l_1^N, \mathcal{M}) p(l_1^N|\mathcal{M}), \tag{4.3}$$

where the sum is over all possible segmentations, and

$$p(\boldsymbol{O}|l_1^N, \mathcal{M}) = \prod_{i=1}^{N} p\big(\boldsymbol{o}_{t(i-1)+1}^{t(i)}|l_i, s_i\big) = \prod_{i=1}^{N} b_i\big(\boldsymbol{o}_{t(i-1)+1}^{t(i)}\big), \tag{4.4}$$

$$p(l_1^N|\mathcal{M}) = p(l_N|s_N) \prod_{i=1}^{N-1} p(l_i|s_i) a_{i,i+1} = d_N(l_N) \prod_{i=1}^{N-1} d_i(l_i) a_{i,i+1}, \tag{4.5}$$

where $t(i)$ is the ending time of the ith segment, $l_i = t(i) - t(i-1)$ the ith segment length or the state duration for state s_i, and d_i the state duration distribution for state s_i. The term $b_i\big(\boldsymbol{o}_{t(i-1)+1}^{t(i)}\big)$ can be calculated by using Equation 4.2 applied to the segment $\boldsymbol{o}_{t(i-1)+1}^{t(i)}$.

4.3 The segmental Viterbi decoder

The segmental Viterbi decoder is used in both estimation of the MSHMM model parameters and recognition. Before going into the detail of the segmental Viterbi decoder, it is worth having a brief review of the Viterbi algorithm for conventional HMMs for comparison.

4.3.1 The Viterbi algorithm for conventional HMMs

Given a sequence of T acoustic observations $\boldsymbol{O} = \{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_T\}$ and a conventional HMM M of N states $\{s_1, \ldots, s_N\}$, a state sequence $\hat{x}_1^T = \{\hat{x}_1, \ldots, \hat{x}_T\}$ of length T is said to be the *optimal state sequence* if the probability of $\boldsymbol{O}$ generated by M via the state sequence $\hat{x}_1^T$ is maximal over all possible state sequences $\{x_1^T\}$ of length T

$$\hat{x}_1^T = \arg\max_{x_1^T} p(\boldsymbol{O}, x_1^T|M). \tag{4.6}$$

The probability $p(\boldsymbol{O}|M)$ of $\boldsymbol{O}$ generated by M can then be approximated by

$$\hat{p}(\boldsymbol{O}|M) = \max_{x_1^T} p(\boldsymbol{O}, x_1^T|M) = p(\boldsymbol{O}, \hat{x}_1^T|M). \tag{4.7}$$

The standard solution in current ASR systems to compute the optimal state sequence $\hat{x}_1^T$ and the corresponding probability $\hat{p}(\boldsymbol{O}|M)$ is the Viterbi algorithm (Viterbi 1967, Forney 1973). Viterbi decoding is a breadth-first search scheme in which all candidate hypotheses are pursued in parallel. Define

$$\hat{\alpha}_i(t) = \max_{x_1,\ldots,x_t} p(\boldsymbol{o}_1, \ldots, \boldsymbol{o}_t, x_1, \ldots, x_t, x_t = s_i|M) \tag{4.8}$$

as the joint probability of the partial observation sequence $\{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_t\}$ and the *partial* optimal state sequence $\tilde{x}_1^t = \{\tilde{x}_1, \ldots, \tilde{x}_t\}$ and the final observation in this sequence, $\boldsymbol{o}_t$, is generated by state s_i, i.e., $x_t = s_i$. It follows that $\hat{\alpha}_i(t)$ can be computed efficiently using the following recursive equation

$$\hat{\alpha}_i(t) = \begin{cases} \pi_i b_i(\boldsymbol{o}_1) & \text{for } t = 1, \\ \max\limits_j \big(\hat{\alpha}_j(t-1) a_{ji} b_i(\boldsymbol{o}_t)\big) & \text{for } t > 1, \end{cases} \tag{4.9}$$

where a_{ji} is the state transition probability from state s_j to state s_i, π_i the initial state probability, and $b_i(\boldsymbol{o}_t)$ the output PDF for state s_i, evaluated at $\boldsymbol{o}_t$. Apparently, the probability $\hat{p}(\boldsymbol{O}|M)$ is given by

$$\hat{p}(\boldsymbol{O}|M) = \hat{\alpha}_N(T). \tag{4.10}$$

The optimal state sequence $\hat{x}_1^T$ can be retrieved provided that at each time t and state i, a record is kept of the state j which corresponds to the maximum at time $t - 1$.

4.3.2 The segmental Viterbi decoder

For a segmental HMM $\mathcal{M}$, the principle of the Viterbi algorithm still applies. By analogy with the notation for the Viterbi algorithm used in the case of a conventional HMM, let

$$\hat{\alpha}_i(t) = \max_{x_1,\ldots,x_t} p(\boldsymbol{o}_1, \ldots, \boldsymbol{o}_t, x_1, \ldots, x_t, x_t = s_i, x_{t+1} \neq s_i | \mathcal{M}) \tag{4.11}$$

be the maximum joint probability of the partial acoustic observation sequence $\{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_t\}$ and the partial optimal state sequence $\tilde{x}_1^t = \{\tilde{x}_1, \ldots, \tilde{x}_t\}$ when the final observation in this sequence, $\boldsymbol{o}_t$, is generated by state s_i, i.e., $x_t = s_i$. In addition, the condition $x_{t+1} \neq s_i$ ensures that only segments which are complete at time t are considered.

The segmental version of the Viterbi decoder is given by Russell & Jackson (2005)

$$\hat{\alpha}_i(t) = \max_j \max_\tau \begin{cases} \pi_j b_j(\boldsymbol{o}_1^\tau) & \text{for } t = \tau, \\ \hat{\alpha}_j(t-\tau) a_{ji} b_i(\boldsymbol{o}_{t-\tau+1}^t) & \text{for } t > \tau, \end{cases} \tag{4.12}$$

where π_j is the probability that the state sequence begins in the jth state, $b_i(\boldsymbol{o}_{t-\tau+1}^t)$ the ith segmental state output PDF evaluated at the segment $\boldsymbol{o}_{t-\tau+1}^t$, τ the state duration ($1 \leq \tau \leq \tau_{\max}$), where $\tau_{\max}$ is the maximum state duration. Ideally, $\tau_{\max}$ should be sufficiently large to accommodate all corpora. Unfortunately, this would lead to increased computational complexity. As a compromise, the smallest maximum state duration that will work is chosen, and in practice, this depends on the given corpus. For the TIMIT corpus used in our experimental work, $\tau_{\max}$ is set to 15 frames, which is sufficient to accommodate all TIMIT phones.

In conventional HMMs, in order to compute $\hat{\alpha}_i(t)$, only states which could be occupied at previous time $t-1$ are considered. However, the segmental Viterbi decoder has to take account of all possible segment duration, τ, and the evaluation of the segmental state output probabilities, $b_i(\boldsymbol{o}_{t-\tau+1}^t)$. Figure 4.1 shows the computation of the probability $\hat{\alpha}_i(t)$ in segmental Viterbi decoding based on the state-time trellis. The time complexity of the standard Viterbi algorithm is $\mathrm{O}(N^2T)$, where N and T are the number of states and the sequence length, respectively. The segmental viterbi algorithm has a time complexity of $\mathrm{O}(N^2T\tau_{\max})$

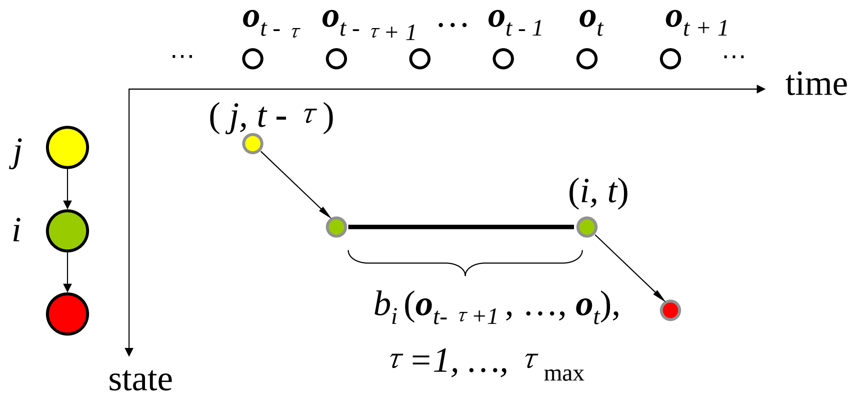

$\boldsymbol{o}_{t-\tau}$
$\boldsymbol{o}_{t-\tau+1}$
...
$\boldsymbol{o}_{t-1}$
$\boldsymbol{o}_{t}$
$\boldsymbol{o}_{t+1}$
time
$(j, t-\tau)$
j
i
(i, t)
$b_i(\boldsymbol{o}_{t-\tau+1}, \ldots, \boldsymbol{o}_t),$
$\tau = 1, \ldots, \tau_{\max}$
state

Figure 4.1: State-time trellis interpretation of the calculation of the probability $\hat{\alpha}_i(t)$ in segmental Viterbi decoding, in which all possible segment durations $\tau \in \{1, \ldots, \tau_{\max}\}$ and the corresponding segmental state output probabilities, $b_i(\boldsymbol{o}_{t-\tau+1}^t)$, should be considered.

or $\mathrm{O}(N^2T^2)$ when there is no restriction on the state duration (i.e., $\tau_{\max} = T$). Russell (2005) proposed two methods to reduce the computational complexity in SHMM decoding for recognition: segmental beam pruning and duration pruning.

Segmental beam pruning

For each time t, let $\hat{i}(t) = \arg\max_i \hat{\alpha}_i(t)$. Given a bean threshold $\xi_b > 0$, state j at time t is pruned if

$$|\log(\hat{\alpha}_{\hat{i}(t)}(t)) - \log(\hat{\alpha}_j(t))| \geq \xi_b. \tag{4.13}$$

The right-hand side of Equation (4.12) is not evaluated at time $t - \tau$ if all its predecessor states j are pruned at $t - \tau$. Moreover, if all state i's predecessors are pruned at time s, then Equation (4.12) is not computed for $t = s + 1, \ldots, s + \tau_{max}$.

Duration pruning

Given a speech segment $\boldsymbol{o}_s^e$ ($1 \leq s < e \leq T$), a duration pruning threshold $\xi_d > 0$ is used in such a way that $b_i(\boldsymbol{o}_{\boldsymbol{s}}^{\mathbf{e}})$ is not evaluated if $d_i(e - s + 1) \leq \xi_d$, where d_i is the state duration distribution for state s_i.

Of course, there is a trade-off between computational cost and recognition performance for segment models. On a context-dependent phone recognition task on TIMIT, for example, Russell (2005) showed that a 95% reduction in the number of segment probability calculations is achieved at the cost of a 3% increase in PER, with $\xi_b = 90$ and $\xi_d = 0.05$.

4.4 Linear/linear MSHMMs parameter estimation

Linear/linear MSHMMs training has two separate stages. Firstly, a set of formant-to-acoustic mappings is estimated, which remain fixed thereafter. Secondly, the other model parameters are estimated based on the formant-to-acoustic mappings learned in the first stage. Section 4.4.1 and Section 4.4.2 describe these two steps respectively.

4.4.1 Estimation of the formant-to-acoustic mappings

The formant-to-acoustic mapping is phone-category-dependent, so that each phone category has a separate mapping. In general, the phone set is partitioned into K categories (the number of categories K varies, depending on the particular phone categorisation schemes). Therefore, a set of K formant-to-acoustic mapping $\{\boldsymbol{W}_1, \ldots, \boldsymbol{W}_K\}$ is obtained. All mappings are linear and estimated using 'matched' sequences of formant and acoustic data for each phone category, as described below.

Let $\boldsymbol{R} = \{\boldsymbol{r}_1, \ldots, \boldsymbol{r}_T\}$ and $\boldsymbol{O} = \{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_T\}$ be matched sets of M-dimensional formant and N-dimensional acoustic vectors corresponding to a particular phone category k. The formant-to-acoustic mapping $\boldsymbol{W}_k$ for this phone category is an $N \times M$ matrix, which is estimated by minimizing the error E between the two matched sets $\boldsymbol{R}$ and $\boldsymbol{O}$

$$E = (\boldsymbol{O} - \boldsymbol{W}_k\boldsymbol{R})^\top \Sigma^{-1} (\boldsymbol{O} - \boldsymbol{W}_k\boldsymbol{R}), \tag{4.14}$$

where $\boldsymbol{O}$ is the $N \times T$ matrix whose tth column is $\boldsymbol{o}_t$, and $\boldsymbol{R}$ is the $M \times T$ matrix whose tth column is $\boldsymbol{r}_t$. $^\top$ and $^{-1}$ denote the transpose and inverse respectively, and $\boldsymbol{\Sigma}$ is the covariance matrix. In the case where $\boldsymbol{\Sigma}$ is assumed to be diagonal, with diagonal elements denoted by

σ_n $(1 \leq n \leq N)$, Equation (4.14) becomes

$$E = \sum_{n=1}^{N} \frac{1}{\sigma_n} \sum_{n=1}^{T} (o_t^n - \boldsymbol{W}_k r_t^n)^2, \tag{4.15}$$

where o_t^n and $\boldsymbol{W}_k r_t^n$ are the n^{th}elements of $\boldsymbol{o}_t$ and $\boldsymbol{W}_k \boldsymbol{r}_t$, respectively. Therefore, minimising E is equivalent to minimising

$$\widetilde{E} = (\boldsymbol{O} - \boldsymbol{W}_k \boldsymbol{R})^\top (\boldsymbol{O} - \boldsymbol{W}_k \boldsymbol{R}). \tag{4.16}$$

This is a standard least-squares problem in linear algebra, with solution

$$\boldsymbol{W}_k = \boldsymbol{O}\boldsymbol{R}^\dagger, \tag{4.17}$$

where $\boldsymbol{R}^\dagger$ is the pseudo-inverse of $\boldsymbol{R}$ (see, e.g., Bishop (1995)) which can be computed by, e.g., Singular Value Decomposition (SVD) (Golub & Kahan 1965, Golub & Loan 1996).

4.4.2 Estimation of the model trajectory parameters

Once the estimation of the formant-to-acoustic mappings is complete, the formant data are discarded and only the acoustic data are used for the estimation of the other model parameters, including the model trajectory parameters (i.e., the midpoint vector $\boldsymbol{c}$ and the slope vector $\boldsymbol{m}$), the covariance matrix $\boldsymbol{\Sigma}$ and the state duration distribution d. The maximum likelihood estimation of these MSHMM parameters are obtained using a version of the EM algorithm based on segmental Viterbi decoding (implemented in SEGVit), including the following main steps:

1. Define an initial MSHMM model set $\mathcal{M}_0$.

2. Compute the optimal state sequence between $\mathcal{M}_0$ and training data $\boldsymbol{O}$ using the segmental Viterbi decoder.

3. Segment $\boldsymbol{O}$ using the optimal state sequence.

4. Re-estimate model parameters based on the segmentation to give a new model set $\mathcal{M}_1$, such that $p(\boldsymbol{O}|\mathcal{M}_0) \leq p(\boldsymbol{O}|\mathcal{M}_1)$.

5. Replace $\mathcal{M}_0$ with $\mathcal{M}_1$ and repeat from steps 2.

The whole process results in a sequence of models $\mathcal{M}_0, \dots, \mathcal{M}_n$ in such a way that $p(\boldsymbol{O}|\mathcal{M}_0) \leq p(\boldsymbol{O}|\mathcal{M}_1) \leq \dots \leq p(\boldsymbol{O}|\mathcal{M}_n)$. This process terminates when the difference $|p(\boldsymbol{O}|\mathcal{M}_n) - p(\boldsymbol{O}|\mathcal{M}_{n-1})|$ falls below some threshhold ϵ (and possibly remains below ϵ over some pre-determined time interval), or a maximum number of iterations is reached. The rest of this section lists various fomulae employed for the model trajectory parameters re-estimation.

MSHMM re-estimation fomulae of the trajectory parameters

The derivation of the re-estimation fomulae for the ML estimations of the midpoint and slope largely follows that in (Russell & Jackson 2005). Firstly, consider a single segment $\boldsymbol{o}_1^{\tau} = \{\boldsymbol{o}_1, \dots, \boldsymbol{o}_{\tau}\}$. The state output probability of the sequence of acoustic feature vectors $\boldsymbol{o}_1^{\tau}$ given state s_j with linear articulatory-to-acoustic mapping $\boldsymbol{W}_k$ and covariance matrix $\boldsymbol{\Sigma}_j$ is given by Equation (4.2) on page 70. This probability can be written as follows according to the definition of a Gaussian PDF

$$b_j(\boldsymbol{o}_1^{\tau}) = \frac{1}{\sqrt{(2\pi)^N |\boldsymbol{\Sigma}_j|}} \prod_{t=1}^{\tau} \exp\Big\{ -\frac{1}{2}\big(\boldsymbol{o}_t - \boldsymbol{W}_k \boldsymbol{f}_j(t)\big)^{\top} \boldsymbol{\Sigma}_j^{-1} \big(\boldsymbol{o}_t - \boldsymbol{W}_k \boldsymbol{f}_j(t)\big)\Big\}, \quad (4.18)$$

where N is the dimensionality of the acoustic observation vector $\boldsymbol{o}_t$. For the multivariate Gaussian densities, maximising the likelihood is equivalent to minimising the Malhalanobis distance E as below

$$E = \sum_{t=1}^{\tau} \Big(\big(\boldsymbol{o}_t - \boldsymbol{W}_k \boldsymbol{f}_j(t)\big)^{\top} \boldsymbol{\Sigma}_j^{-1} \big(\boldsymbol{o}_t - \boldsymbol{W}_k \boldsymbol{f}_j(t)\big)\Big). \quad (4.19)$$

Therefore, the maximum of the likelihood function can be found from the point at which E's gradient is zero.

ML estimation of the midpoint Assume that w_n is the nth row of the mapping $\boldsymbol{W}_k$. We calculate the derivative

$$\begin{aligned}\frac{\partial E}{\partial \boldsymbol{c}_n} &= \sum_{t=1}^{\tau} \frac{\partial}{\partial \boldsymbol{c}_n}\Big(\big(\boldsymbol{o}_t - \boldsymbol{W}_k \boldsymbol{f}_j(t)\big)^\top \boldsymbol{\Sigma}_j^{-1}\big(\boldsymbol{o}_t - \boldsymbol{W}_k \boldsymbol{f}_j(t)\big)\Big)\\ &= 2\sum_{t=1}^{\tau} \mathrm{w}_n \Sigma_j^{-1}\big(\boldsymbol{o}_t - \boldsymbol{W}_k \boldsymbol{f}_j(t)\big). \end{aligned} \tag{4.20}$$

Based on Equation (4.20) and the definition of the trajectory $\boldsymbol{f}_j(t)$ (see Equation 4.1), it can be shown that at the optimum

$$\mathrm{w}_n \Sigma_j^{-1} \sum_{t=1}^{\tau} \Big(\boldsymbol{o}_t - \boldsymbol{W}_k\big(\boldsymbol{m}_j(t-\bar{t}) + \hat{\boldsymbol{c}}_j\big)\Big) = 0. \tag{4.21}$$

Since $\sum_{t=1}^{\tau}(t-\bar{t}) = 0$, the above equation becomes

$$\mathrm{w}_n \Sigma_j^{-1} \sum_{t=1}^{\tau} \boldsymbol{o}_t - \mathrm{w}_n \Sigma_j^{-1} \sum_{t=1}^{\tau} \boldsymbol{W}_k \hat{\boldsymbol{c}}_j = 0. \tag{4.22}$$

From a matrix point of view, Equation (4.22) can be written as

$$\hat{\boldsymbol{c}}_j = \frac{1}{\tau}(\boldsymbol{W}_k^\top \boldsymbol{\Sigma}_j^{-1} \boldsymbol{W}_k)^{-1} \boldsymbol{W}_k^\top \boldsymbol{\Sigma}_j^{-1} \sum_{t=1}^{\tau} \boldsymbol{o}_t = \frac{1}{\tau}(\boldsymbol{\Psi}_j \boldsymbol{W}_k)^\dagger \boldsymbol{\Psi}_j \sum_{t=1}^{\tau} \boldsymbol{o}_t, \tag{4.23}$$

where $\boldsymbol{\Psi}_j = \boldsymbol{\Sigma}_j^{-1/2}$ and † denotes the pseudo-inverse.

ML estimation of the slope Similarly, the ML estimation of the slope $\boldsymbol{m}_j$ can be found by calculating

$$\begin{aligned}\frac{\partial E}{\partial \boldsymbol{m}_n} &= \sum_{t=1}^{\tau} \frac{\partial}{\partial \boldsymbol{m}_n}\Big(\big(\boldsymbol{o}_t - \boldsymbol{W}_k \boldsymbol{f}_j(t)\big)^\top \boldsymbol{\Sigma}_j^{-1}\big(\boldsymbol{o}_t - \boldsymbol{W}_k \boldsymbol{f}_j(t)\big)\Big)\\ &= 2\sum_{t=1}^{\tau} (t-\bar{t})\mathrm{w}_n \Sigma_j^{-1}\big(\boldsymbol{o}_t - \boldsymbol{W}_k \boldsymbol{f}_j(t)\big). \end{aligned} \tag{4.24}$$

At the optimum, the above equation is zero, which yields

$$\mathrm{w}_n \boldsymbol{\Sigma}_j^{-1} \Big(\sum_{t=1}^{\tau} (t-\bar{t}) \boldsymbol{o}_t - \boldsymbol{W}_k \sum_{t=1}^{\tau} \hat{\boldsymbol{m}}_j (t-\bar{t})^2 - \boldsymbol{W}_k \sum_{t=1}^{\tau} \boldsymbol{c}_j (t-\bar{t}) \Big) = 0. \tag{4.25}$$

Again, since $\sum_{t=1}^{\tau}(t-\bar{t}) = 0$, equation (4.25) reduces to

$$\mathrm{w}_n \boldsymbol{\Sigma}_j^{-1} \sum_{t=1}^{\tau} (t-\bar{t}) \boldsymbol{o}_t - \mathrm{w}_n \boldsymbol{\Sigma}_j^{-1} \boldsymbol{W}_k \sum_{t=1}^{\tau} \hat{\boldsymbol{m}}_j (t-\bar{t})^2 = 0. \tag{4.26}$$

As before, the above equation can be re-arranged in the following matrix form as

$$\begin{aligned} \hat{\boldsymbol{m}}_j &= \frac{1}{\sum\limits_{t=1}^{\tau}(t-\bar{t})^2} (\boldsymbol{W}_k^\top \boldsymbol{\Sigma}_j^{-1} \boldsymbol{W}_k)^{-1} \boldsymbol{W}_k^\top \boldsymbol{\Sigma}_j^{-1} \sum_{t=1}^{\tau} (t-\bar{t}) \boldsymbol{o}_t \\ &= \frac{1}{\sum\limits_{t=1}^{\tau}(t-\bar{t})^2} (\boldsymbol{\Psi}_j \boldsymbol{W}_k)^\dagger \boldsymbol{\Psi}_j \sum_{t=1}^{\tau} (t-\bar{t}) \boldsymbol{o}_t. \end{aligned} \tag{4.27}$$

Recall that in the above equations, $\bar{t} = (\tau+1)/2$ is the midpoint of the start and end time points. Equations (4.23) and (4.27) are just for a single segment. Now suppose that we have a training observation sequence $\boldsymbol{O} = \{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_T\}$ of length T and the corresponding optimal state sequence $\hat{x}_1^T$. Let T_j be the total number of frames spent in state s_j and J_j be the number of occurrences of the state s_j in $\hat{x}_1^T$. Suppose that t_i^s and t_i^e are the start and end times for the ith occurrence of state s_j in $\hat{x}_1^T$, respectively, and $\bar{t}_i = (t_i^s + t_i^e)/2$ is the midpoint of the two points t_i^s and t_i^e. In this case, it can be shown that the ML estimates of the midpoint $\hat{\boldsymbol{c}}_j$ and slope $\hat{\boldsymbol{m}}_j$ for state s_j are given by:

$$\hat{\boldsymbol{c}}_j = \frac{1}{T_j} \sum_{i=1}^{J_j} \sum_{t=t_i^s}^{t_i^e} (\boldsymbol{\Psi}_j \boldsymbol{W}_k)^\dagger \boldsymbol{\Psi}_j \boldsymbol{o}_t, \tag{4.28}$$

$$\hat{\boldsymbol{m}}_j = \frac{\sum\limits_{i=1}^{J_j} \sum\limits_{t=t_i^s}^{t_i^e} (t-\bar{t}_i) (\boldsymbol{\Psi}_j \boldsymbol{W}_k)^\dagger \boldsymbol{\Psi}_j \boldsymbol{o}_t}{\sum\limits_{i=1}^{J_j} \sum\limits_{t=t_i^s}^{t_i^e} (t-\bar{t}_i)^2}. \tag{4.29}$$

In summary, given an annotated training utterance and a force-aligned optimal state sequence computed using the segmental Viterbi algorithm, the difference between the transformed state trajectory values $\boldsymbol{W}_k(\boldsymbol{f}_j)$ and the actual feature vectors are used to estimated the covariance Σ_j in the acoustic domain. On the other hand, an acoustic segment corresponding to a state j is then 'pulled back' (Russell et al. 2007) into the articulatory domain using the pseudo-inverse matrix $\boldsymbol{W}_k^\dagger$, and used to estimate new values for the trajectory midpoint and slope parameters. As can be seen in Equations (4.28) and (4.29), the ML estimates of the trajectory parameters in the articulatory domain are those which give the best linear fit to the (pseudo) inverse-transformed acoustic observation vectors, taking account of the covariance information in the matrix $\boldsymbol{\Psi}_j$.

4.5 Previous experimental work and main findings

In the previous experimental work on a linear/linear MSHMM (Russell & Jackson 2005, Russell et al. 2007), the following three types of intermediate representation are investigated:

- the first 3 formant frequencies (3FF)
- first 3 formant frequencies plus 5 frequency-band energy (3FF+5BE)
- the 12 PFS control parameters from HMS formant synthesiser (12PFS)

and these parameters have been described in Section 3.2.3 on page 59, when formant anaysis is reviewed. In addition, five formant-to-acoustic mapping schemes have been studied, as shown in Table 4.1.[1] They are referred to as 1A, 6B, 10C, 10D and 49E, respectively. The letters A, B, C, D and E denote the five different mapping schemes, and the numbers preceding them indicate the number of distinct mappings within each mapping scheme.

Linear/linear MSHMMs phone *classification* results on the TIMIT corpus were reported in (Russell & Jackson 2005), which were limited to male subjects of the TIMIT corpus in order to reduce variability caused by different gender of speakers. Experiments were conducted using both CI (using the TIMIT 49-phone set, see appdendix A) and CD MSHMMs

[1] These categorisations of the phones are suggested by Jackson et al. (2002).

1A	all phones
6B	vowels, {hh, l, r, w, y}, nasals, {dh, f, s, sh, th, v, z, zh} {ch, jh}, {b, cl, d, vcl, dx, epi, g, k, p, q, sil, t}
10C	vowels, {epi, q, sil}, {hh, l, r, w, y}, nasals, {ch, s} {sh, f, th}, {dh, v, zh}, {jh, z}, {cl, k, p, t}, {b, d, vcl, dx, g}
10D	vowels, {epi, q, sil}, {dx, el, l, r, w, y}, nasals, {vcl}, {cl} {b, d, g, jh}, {ch, k, p, q, t}, {dh, v, z, zh}, {f, hh, s, sh, th}
49E	49 individual phones

Table 4.1: The five formant-to-acoustic mapping schemes used in the linear/linear MSHMM experimental framework. B. linguistic categories; C. as in (Deng & Ma, 2000); D. discrete articulatory regions. 'nasals' and 'vowels' denote the sets {en, m, n, ng}, and {aa, ae, ah, ao, aw, ax, ay, eh, el, er, ey, ih, ix, iy, ow, oy, uh, uw}, respectively. TIMIT symbols are used in this table.

(1400 triphones), with or without a language model. Three formant-based intermediate representations and five formant-to-acoustic mapping schemes were considered. The baseline phone classification experiments were conducted using FTSHMMs, with no intermediate layer (Holmes & Russell 1999, Jackson & Russell 2002). This type of linear trajectory segmental HMMs (i.e., linear FTSHMMs, as discussed in detail in Section 2.3.4) provides the theoretical upper bound for the performance of all the linear/linear MSHMMs systems. (Russell & Jackson (2005) showed how a linear/linear MSHMM is derived from a linear FTSHMM, in which a linear trajectory in the acoustic domain was realised in the intermediate layer and transformed into the acoustic layer using a linear mapping.) The main findings of previous experimental work are summarised as follows.

- In general, the best phone classification accuracy is obtained by increasing either the dimension of the intermediate layer or the number of articulatory-to-acoustic mappings.

- The theoretical upper bound can be achieved, provided that the intermediate layer is sufficiently rich, or the articulatory-to-acoustic mapping is sufficiently flexible.

- It is demonstrated that the triphone MSHMM system with the 3FF intermediate representation and 49 articulatory-to-acoustic mappings has 25% fewer parameters and gives superior performance, compared with the conventional HMM system.

Russell et al. (2007) continued to work on linear/linear MSHMMs. This time, they presented phone *recognition* results for both male and female speakers using triphone MSHMMs (the number of triphone models for male and female speakers are 1364 and 660 respectively) with a bigram language model on the TIMIT corpus. The experiments were divided into two categories, within-gender and cross-gender experiments. Again, the baseline phone recognition experiments were conducted using linear trajectory FTSHMMs, with no intermediate layer. The results of within-gender experiments followed the same trend as the phone classification results reported in (Russell & Jackson 2005). In addition, it is found that there is no evidence that performance for female speech is affected by difficulties in formant analysis for female speakers.

4.6 Summary

This chapter gives a detailed account of the MSHMM framework, which this thesis work builds upon. The theory of a linear/linear MSHMM is reviewed carefully, including model components, the segmental Viterbi decoding algorithm and the model parameter estimation algorithm. At the end, this chapter describes the previous experimental framework, highlighting some of the important experiment results for a linear/linear MSHMM.

Within the general MSHMM framework, there is certainly room for development of many kinds. As has been discussed in Chapter 1, one of the limitations of a linear/linear MSHMM is the use of piecewise linear trajectories to model formant dynamics. The goal of this research is to extend the linear/linear MSHMM to incorporate an improved, non-linear trajectory model of formant dynamics. Of course, there are many ways to generate smoothed non-linear trajectories, as described in Chapter 2. However, the particular approach adopted in this research is the SPG algorithm (Tokuda, Kobayashi & Imai 1995), which is detailed in the next chapter.

Chapter 5

Non-linear trajectories generation

5.1 Introduction

This chapter first describes the technique used in this research to generate the non-linear trajectory for modelling formant dynamics, i.e., the speech parameter generation algorithm (Tokuda, Kobayashi & Imai 1995), then goes on to discuss the trajectory HMM method (Tokuda et al. 2003) since the formulation of the trajectory HMM is closely related to the SPG technique. These two algorithms are described in Sections 5.2 and 5.3, respectively. One of the most successful applications of the SPG algorithm is HMM-based speech synthesis, which is outlined in Section 5.4.

5.1.1 Motivation behind Tokuda's algorithms

In Section 2.2.3, the use of dynamic features in HMMs to account for temporal feature dynamics is reviewed. Despite its practical benefits in improving performance for a wide range of applications, the use of dynamic features results in inconsistency in HMM assumptions. In a conventional HMM, it is assumed that both the static and dynamic parameters are simultaneously constant and possibly non-zero within a state occupancy. However, the constant static parameters within states should have caused all delta parameters to be zero. On the other hand, the non-zero delta parameters within a state occupancy should have resulted in time-varying static parameters.

In recognition of both the popularity of the use of dynamic features in speech pattern modelling and the inconsistency caused by the inclusion of these dynamics parameters in standard HMMs, Tokuda and colleagues have been working on a principled approach which aims at alleviating the inconsistency introduced by the use of dynamic features in conventional HMMs. As a starting point, Tokuda, Kobayashi & Imai (1995) began their research on seeking a method to 'synthesise' a smoothed, non-linear speech parameter sequence from HMMs which include both static and dynamic features. The key motive behind this algorithm is that the speech parameter sequence sought after should be regulated by the constraints between static and dynamic features. Otherwise, a sequence of piecewise constant mean vectors is generated. The SPG algorithm in effect performs an utterance-level smoothing on the piecewise constant mean trajectory, taking account of the constraints between static and dynamic features, together with the covariance information. As a result, a smoothed, non-linear trajectory which is most consistent with static and dynamic constraints is synthesised in the static feature space.

The generated non-linear speech parameter trajectory was initially used in HMM-based speech synthesis (Masuko, Tokuda, Kobayashi & Imai 1996), which can significantly improve the quality of the synthetic speech. Minami, McDermott, Nakamura & Katagiri (2002) also applied this speech parameter generation technique to speech recognition by rescoring an N-best list, with positive results (18.2% reduction in word error rate) obtained on a speaker-independent recognition task relative to HMMs. Later work (Tokuda et al. 2003) found that the standard HMM can be transformed into a new trajectory model – namely, the trajectory HMM – by assuming the non-linear speech parameter trajectory to be the mean for the static observation sequence corresponding to an utterance. Such an approach can overcome the limitations of the piecewise stationarity and independence assumptions in HMMs.

5.1.2 Relationships between static and dynamic features

Before going into the detail of Tokuda's algorithms, it is useful to examine the relationships between static and dynamic features. Let $\boldsymbol{O} = \{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_T\}$ be a sequence of T speech observations (e.g., mel-cepstra). Assume that each speech vector $\boldsymbol{o}_t$ consists of both a static feature vector $\boldsymbol{c}_t$ and dynamic feature vectors $\Delta\boldsymbol{c}_t$, $\Delta^2\boldsymbol{c}_t$. The delta and delta-delta coefficients can be computed using Equation (2.2) on page 20, which is re-stated as below for completeness.

$$\begin{aligned}\Delta\boldsymbol{c}_t &= \frac{\sum\limits_{\theta=1}^{\Theta}\theta(\boldsymbol{c}_{t+\theta}-\boldsymbol{c}_{t-\theta})}{2\sum\limits_{\theta=1}^{\Theta}\theta^2}, && (5.1)\\ \Delta^2\boldsymbol{c}_t &= \frac{\sum\limits_{\theta=1}^{\Theta}\theta(\Delta\boldsymbol{c}_{t+\theta}-\Delta\boldsymbol{c}_{t-\theta})}{2\sum\limits_{\theta=1}^{\Theta}\theta^2}, && (5.2)\end{aligned}$$

where Θ determines the length of the regression window. Suppose that the static feature vector $\boldsymbol{c}_t$ is of dimension D (therefore, $\boldsymbol{o}_t$ is of dimension $3D$). Note that the calculation of Δ and Δ^2 parameters at any time t requires a number of preceding and following speech feature vectors, so it is necessary to make some adjustment to the above fomulae when computing Δ and Δ^2 at the beginning and end of the feature vector sequence. A simple solution to this end-effect problem is to replicate the first or last feature vector as many times as required.

The expanded observation sequence $\boldsymbol{O}$ and the static observation sequence $\boldsymbol{C}$ can be rewritten as:

$$\boldsymbol{O} = [\boldsymbol{o}_1, \ldots, \boldsymbol{o}_T]^\top, \tag{5.3}$$

$$\boldsymbol{C} = [\boldsymbol{c}_1, \ldots, \boldsymbol{c}_T]^\top, \tag{5.4}$$

where

$$\boldsymbol{c}_t = [c_t(1), \ldots, c_t(D)], \tag{5.5}$$

$$\boldsymbol{o}_t = \left[\boldsymbol{c}_t, \Delta\boldsymbol{c}_t, \Delta^2\boldsymbol{c}_t\right]. \tag{5.6}$$

Assume that all vectors are row vectors by default. So $\boldsymbol{O} = [\boldsymbol{o}_1, \ldots, \boldsymbol{o}_T]^\top$ becomes a $3DT \times 1$ column vector and $\boldsymbol{C}$ becomes a $DT \times 1$ column vector. Now define $\boldsymbol{W}$ as a linear mapping which transforms the sequence of static feature vectors $\boldsymbol{C}$ into the augmented form $\boldsymbol{O}$, which includs both static and dynamic feature. Then the relationship between $\boldsymbol{O}$ and $\boldsymbol{C}$ can be represented in a matrix form

$$\boldsymbol{O} = \boldsymbol{W}\boldsymbol{C}. \tag{5.7}$$

The mapping $\boldsymbol{W}$ is a $3DT \times DT$ matrix which can be expanded as below

$$\boldsymbol{W} = \left[\begin{array}{ccccccccccccc}
\boldsymbol{I} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \dots & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} \\
-\frac{1}{2}\boldsymbol{I} & \frac{1}{2}\boldsymbol{I} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \dots & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} \\
-\frac{1}{4}\boldsymbol{I} & \boldsymbol{O} & \frac{1}{4}\boldsymbol{I} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \dots & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} \\
\hline
\boldsymbol{O} & \boldsymbol{I} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \dots & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} \\
-\frac{1}{2}\boldsymbol{I} & \boldsymbol{O} & \frac{1}{2}\boldsymbol{I} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \dots & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} \\
\frac{1}{4}\boldsymbol{I} & -\frac{1}{2}\boldsymbol{I} & \boldsymbol{O} & \frac{1}{4}\boldsymbol{I} & \boldsymbol{O} & \boldsymbol{O} & \dots & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} \\
\hline
\boldsymbol{O} & \boldsymbol{O} & \boldsymbol{I} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \dots & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} \\
\boldsymbol{O} & -\frac{1}{2}\boldsymbol{I} & \boldsymbol{O} & \frac{1}{2}\boldsymbol{I} & \boldsymbol{O} & \boldsymbol{O} & \dots & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} \\
\frac{1}{4}\boldsymbol{I} & \boldsymbol{O} & -\frac{1}{2}\boldsymbol{I} & \boldsymbol{O} & \frac{1}{4}\boldsymbol{I} & \boldsymbol{O} & \dots & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} \\
\hline
\multicolumn{13}{c}{\ddots} \\
\hline
\boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \dots & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{I} & \boldsymbol{O} & \boldsymbol{O} \\
\boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \dots & \boldsymbol{O} & \boldsymbol{O} & -\frac{1}{2}\boldsymbol{I} & \boldsymbol{O} & \frac{1}{2}\boldsymbol{I} & \boldsymbol{O} \\
\boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \dots & \boldsymbol{O} & \frac{1}{4}\boldsymbol{I} & \boldsymbol{O} & -\frac{1}{2}\boldsymbol{I} & \boldsymbol{O} & \frac{1}{4}\boldsymbol{I} \\
\hline
\boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \dots & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{I} & \boldsymbol{O} \\
\boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \dots & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & -\frac{1}{2}\boldsymbol{I} & \boldsymbol{O} & \frac{1}{2}\boldsymbol{I} \\
\boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \dots & \boldsymbol{O} & \boldsymbol{O} & \frac{1}{4}\boldsymbol{I} & \boldsymbol{O} & -\frac{1}{2}\boldsymbol{I} & \frac{1}{4}\boldsymbol{I} \\
\hline
\boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \dots & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{I} \\
\boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \dots & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & -\frac{1}{2}\boldsymbol{I} & \frac{1}{2}\boldsymbol{I} \\
\boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \dots & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \frac{1}{4}\boldsymbol{I} & \boldsymbol{O} & -\frac{1}{4}\boldsymbol{I}
\end{array}\right],$$

where $\boldsymbol{I}$ and $\boldsymbol{O}$ denote the $D \times D$ identity matrix and $D \times D$ zero matrix, respectively. For the illustration of $\boldsymbol{W}$, Θ in Equations (5.1, 5.2) is set at the value of 1. The top block of $\boldsymbol{W}$, over the first horizontal line, contributes to the computation of $\Delta\boldsymbol{c}_1$ and $\Delta^2\boldsymbol{c}_1$. The second block (between the first and second horizontal lines) is used to compute $\Delta\boldsymbol{c}_2$ and $\Delta^2\boldsymbol{c}_2$. In general, the t^{th} block is used to compute $\Delta \boldsymbol{c}_t$ and $\Delta^2 \boldsymbol{c}_t$. The end-effect is also shown in $\boldsymbol{W}$ when calculating the first two and final two speech feature vectors $\Delta \boldsymbol{c}_1$, $\Delta^2 \boldsymbol{c}_1$, $\Delta \boldsymbol{c}_2$, $\Delta^2 \boldsymbol{c}_2$ and $\Delta \boldsymbol{c}_{T-1}$, $\Delta^2 \boldsymbol{c}_{T-1}$, $\Delta \boldsymbol{c}_T$, $\Delta^2 \boldsymbol{c}_T$. Note the difference between the top and bottom two blocks and the rest of the blocks in $\boldsymbol{W}$.

5.2 The speech parameter generation algorithms

Several versions of the SPG algorithm have been developed to solve different problems (Tokuda, Kobayashi & Imai 1995, Tokuda, Masuko, Yamada, Kobayashi & Imai 1995, Tokuda et al. 2000). The speech parameter generation problems were summarised in (Tokuda et al. 2000) as:

1. given an HMM $\mathcal{M}$ and a state sequence $x = \{x_1, \ldots, x_T\}$, maximize $p(\boldsymbol{O}|x, \mathcal{M})$ with respect to $\boldsymbol{O}$.

2. given an HMM $\mathcal{M}$, maximize $p(\boldsymbol{O}, x|\mathcal{M})$ with respect to $\boldsymbol{O}$ and x.

3. given an HMM $\mathcal{M}$, maximize $p(\boldsymbol{O}|\mathcal{M})$ with respect to $\boldsymbol{O}$.

For all three problems, the constraint $\boldsymbol{O} = \boldsymbol{W}\boldsymbol{C}$ is imposed. The first two problems were dealt with in (Tokuda, Kobayashi & Imai 1995, Tokuda, Masuko, Yamada, Kobayashi & Imai 1995) and the third problem was discussed in (Tokuda et al. 2000).

The first case is at the heart of the SPG algorithms because the other two cases are built on the first one. This is also the one applied to the research presented in this thesis. In the first SPG problem, it is assumed that a state sequence $x = \{x_1, \ldots, x_T\}$ is given. In other words, the problem is to find a sequence of T static speech parameters $\hat{\boldsymbol{C}}$, such that

$$\hat{\boldsymbol{C}} = \arg\max_{\boldsymbol{C}} p(\boldsymbol{O}|x, \mathcal{M}) = \arg\max_{\boldsymbol{C}} p(\boldsymbol{W}\boldsymbol{C}|x, \mathcal{M}) \tag{5.8}$$

under the constraint of $\boldsymbol{O} = \boldsymbol{W}\boldsymbol{C}$.

Assume that each state of $\mathcal{M}$ is associated with a single Gaussian PDF (the Gaussian mixture version of the speech parameter generation algorithm was described in (Tokuda, Masuko, Yamada, Kobayashi & Imai 1995, Tokuda et al. 2000)). Let $\boldsymbol{\mu}_{x_t}$ and $\boldsymbol{\Sigma}_{x_t}$ be the $1 \times 3D$ mean vector and $3D \times 3D$ covariance matrix associated with state x_t. $\mathcal{N}(\boldsymbol{o}_t|\boldsymbol{\mu}_{x_t}, \boldsymbol{\Sigma}_{x_t})$ denotes the multivariate Gaussian PDF, with mean $\boldsymbol{\mu}_{x_t}$ and diagonal covariance $\boldsymbol{\Sigma}_{x_t}$, evaluated at $\boldsymbol{o}_t$. The probability $p(\boldsymbol{O}|x, \mathcal{M})$ of a sequence of T observations $\boldsymbol{O} = \{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_T\}$ generated by the HMM $\mathcal{M}$ via the state sequence x is given by

$$p(\boldsymbol{O}|x, \mathcal{M}) = \prod_{t=1}^{T} \mathcal{N}(\boldsymbol{o}_t|\boldsymbol{\mu}_{x_t}, \boldsymbol{\Sigma}_{x_t}) = \mathcal{N}(\boldsymbol{O}|\boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x), \tag{5.9}$$

where

$$\boldsymbol{\mu}_x = [\boldsymbol{\mu}_{x_1}, \ldots, \boldsymbol{\mu}_{x_T}]^\top, \tag{5.10}$$

$$\boldsymbol{\Sigma}_x = \operatorname{diag}[\boldsymbol{\Sigma}_{x_1}, \ldots, \boldsymbol{\Sigma}_{x_T}]. \tag{5.11}$$

Taking the the logarithm of $p(\boldsymbol{O}|x, \mathcal{M})$, we have

$$\log p(\boldsymbol{O}|x, \mathcal{M}) = -\frac{1}{2}\boldsymbol{O}^\top \boldsymbol{\Sigma}_x^{-1} \boldsymbol{O} + \boldsymbol{O}^\top \boldsymbol{\Sigma}_x^{-1} \boldsymbol{\mu}_x + K_x, \tag{5.12}$$

where

$$\boldsymbol{\Sigma}_x^{-1} = \operatorname{diag}[\boldsymbol{\Sigma}_{x_1}^{-1}, \ldots, \boldsymbol{\Sigma}_{x_T}^{-1}], \tag{5.13}$$

and K_x is a normalisation constant which is independent of $\boldsymbol{O}$ (see Equation 5.26). Obviously, $p(\boldsymbol{O}|x, \mathcal{M})$ is maximised with respect to $\boldsymbol{O}$ when $\boldsymbol{O} = \boldsymbol{\mu}_x$ without the constraint of $\boldsymbol{O} = \boldsymbol{W}\boldsymbol{C}$; that is, the speech parameter vector sequence becomes a sequence of the mean vectors.

Now taking into account the relationships between $\boldsymbol{O}$ and $\boldsymbol{C}$ (Equation 5.7), maximising the probability $p(\boldsymbol{O}|x, \mathcal{M})$ with respect to the augmented observation sequence $\boldsymbol{O}$ is equivalent to maximising it with respect to the static observation sequence $\boldsymbol{C}$. By expanding $\log p(\boldsymbol{O}|x, \mathcal{M}) = \log p(\boldsymbol{W}\boldsymbol{C}|x, \mathcal{M})$ according to the definition of a Gaussian distribution, differentiating with respect to $\boldsymbol{C}$, and setting the partial derivative to zero

$$\frac{\partial \log p(\boldsymbol{W}\boldsymbol{C}|x, \mathcal{M})}{\partial \boldsymbol{C}} = \mathbf{0}, \tag{5.14}$$

a set of linear equations are obtained, which can be written as

$$\boldsymbol{R}_x \boldsymbol{C} = \boldsymbol{r}_x, \tag{5.15}$$

where

$$\boldsymbol{R}_x = \boldsymbol{W}^\top \boldsymbol{\Sigma}_x^{-1} \boldsymbol{W}, \tag{5.16}$$

$$\boldsymbol{r}_x = \boldsymbol{W}^\top \boldsymbol{\Sigma}_x^{-1} \boldsymbol{\mu}_x. \tag{5.17}$$

The new static feature vector sequence $\hat{\boldsymbol{C}}$, which maximizes $p(\boldsymbol{O}|x, \mathcal{M})$ under the constraints given by $\boldsymbol{O} = \boldsymbol{W}\boldsymbol{C}$, can now be obtained by solving Equation (5.15).

The second speech parameter generation problem attempts to maximise $p(\boldsymbol{O}, x|\mathcal{M}) = p(\boldsymbol{O}|x, \mathcal{M})P(x|\mathcal{M})$ with respect to $\boldsymbol{O}$ under the constraint $\boldsymbol{O} = \boldsymbol{W}\boldsymbol{C}$, taking account of all possible state sequence x. This is computationally impractical due to the huge number of possible state sequences. However, if the optimal state sequence is known, this problem is equivalent to maximising the probability $p(\boldsymbol{O}|x, \mathcal{M})$ with respect to $\boldsymbol{O}$, which reduces the first speech parameter generation problem. The optimal or sub-optimal state sequence can be derived using the standard Viterbi algorithm or the algorithm described in (Tokuda, Kobayashi & Imai 1995, Tokuda, Masuko, Yamada, Kobayashi & Imai 1995). Given the optimal state sequence, the speech parameter sequence can be obtained using the algorithm developed for case 1.

As for the third case, the problem is to maximize $p(\boldsymbol{O}|\mathcal{M})$ with respect to $\boldsymbol{O}$. Tokuda et al. (2000) proposed an EM-type algorithm to find the critical point of the likelihood function $p(\boldsymbol{O}|\mathcal{M})$. In this algorithm, the auxiliary function of the current speech parameter sequence $\boldsymbol{O}$ and a new parameter vector sequence $\boldsymbol{O}^{'}$ is given by

$$\mathcal{Q}(\boldsymbol{O}, \boldsymbol{O}^{'}) = \sum_{\text{all } x} p(\boldsymbol{O}, x|\mathcal{M}) \log p(\boldsymbol{O}^{'}, x|\mathcal{M}). \tag{5.18}$$

Tokuda et al. (2000) showed that by substituting $\boldsymbol{O}^{'}$ which maximises the $\mathcal{Q}$ function (5.18) for $\boldsymbol{O}$, the likelihood increases unless $\boldsymbol{O}$ is a critical point of the likelihood. Under the constraint $\boldsymbol{O}^{'} = \boldsymbol{W}\boldsymbol{C}^{'}$, $\boldsymbol{C}^{'}$ (i.e., the static parameter sequence of $\boldsymbol{O}^{'}$) which maximises $\mathcal{Q}(\boldsymbol{O}, \boldsymbol{O}^{'})$ can be obtained by solving a set of equations which has the same form of Equation (5.15). As a result, the solution for the third speech parameter generation problem involves iterations of the forward-backward algorithm and the speech parameter generation algorithm for the case that the state sequence is known.

Having outlined the three types of SPG algorithms, it can be seen that different algorithms target different problems. The choice of the particular SPG algorithm is therefore up to the specific application at hand and should be chosen carefully. As far as this research is concerned, the goal is to develop and assess an improved, non-linear trajectory model, relative to a piecewise linear trajectory model as used in previous MSHMMs. Therefore, it is the non-linearity of the algorithm that is of greatest interest to this work. Among the three SPG algorithms, only the original SPG algorithm, i.e., the first one, actually performs the non-linearization. The other two algorithms both revolve around the first algorithm. In all cases the non-linear speech parameter sequence is generated by using the first algorithm for which the state sequence is known. For these reasons, the first SPG algorithm is most suitable for this research and therefore is chosen for this thesis work.

5.3 The trajectory HMM algorithm

Given a standard HMM $\mathcal{M}$ and a state sequence $x = \{x_1, \ldots, x_T\}$ of length T, a new trajectory $\bar{\boldsymbol{C}}_x$ is calculated in the static feature space, which provides the best fit under the static and dynamic constraints. Assume that this new trajectory $\bar{\boldsymbol{C}}_x$ is the mean vectors, the standard HMM $\mathcal{M}$ can be transformed into a trajectory model, denoted as $\bar{\mathcal{M}}$, for which the output probability of an observation sequence $\boldsymbol{O}$ becomes the function of the corresponding static feature sequence $\boldsymbol{C}$. The relationship between the standard HMM $\mathcal{M}$ and the corresponding trajectory HMM $\bar{\mathcal{M}}$ is given by

$$p(\boldsymbol{O}|x, \mathcal{M}) = K_x p(\boldsymbol{C}|x, \bar{\mathcal{M}}), \tag{5.19}$$

where K_x is a state-sequence-dependent normalisation constant in order to give a valid PDF. In case that each state output PDF in the standard HMM $\mathcal{M}$ is Gaussian with a diagonal covariance, the new trajectory HMM $p(\boldsymbol{C}|x, \bar{\mathcal{M}})$ is also Gaussian.

5.3.1 The formulation of the trajectory HMM

The formulation of the trajectory HMM is closely related to the SPG algorithm described in Section 5.2. As will be seen, many of the equations and results are identical. In a standard HMM $\mathcal{M}$, the probability of a sequence of speech observation vectors $\boldsymbol{O} = \{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_T\}$ given $\mathcal{M}$ is

$$p(\boldsymbol{O}|\mathcal{M}) = \sum_x p(\boldsymbol{O}|x, \mathcal{M}) p(x|\mathcal{M}), \tag{5.20}$$

where the sum is taken for all possible state sequences x of length T. Assume that the speech feature vector $\boldsymbol{o}_t$ at time t consists a static feature vector $\boldsymbol{c}_t$ and dynamic feature vectors $\Delta\boldsymbol{c}_t$ and $\Delta^2\boldsymbol{c}_t$. The relationship between the static features and dynamic features is regulated by $\boldsymbol{O} = \boldsymbol{W}\boldsymbol{C}$ and the probability $p(\boldsymbol{O}|x, \mathcal{M})$ is given by Equation (5.9).

Substituting $\boldsymbol{O} = \boldsymbol{W}\boldsymbol{C}$ for Equation (5.9)

$$p(\boldsymbol{O}|x, \mathcal{M}) = p(\boldsymbol{W}\boldsymbol{C}|x, \mathcal{M}) = \mathcal{N}(\boldsymbol{W}\boldsymbol{C}|\boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x) = K_x \mathcal{N}(\boldsymbol{C}|\bar{\boldsymbol{C}}_x, \boldsymbol{P}_x), \tag{5.21}$$

where $\boldsymbol{\mu}_x$ and $\boldsymbol{\Sigma}_x$ are given by Equations (5.10,5.11), respectively and $\bar{\boldsymbol{C}}_x$ is given by

$$\boldsymbol{R}_x \bar{\boldsymbol{C}}_x = \boldsymbol{r}_x. \tag{5.22}$$

$\boldsymbol{P}_x$, $\boldsymbol{R}_x$ and $\boldsymbol{r}_x$ are given by

$$\boldsymbol{P}_x = \boldsymbol{R}_x^{-1}, \tag{5.23}$$

$$\boldsymbol{R}_x = \boldsymbol{W}^\top \boldsymbol{\Sigma}_x^{-1} \boldsymbol{W}, \tag{5.24}$$

$$\boldsymbol{r}_x = \boldsymbol{W}^\top \boldsymbol{\Sigma}_x^{-1} \boldsymbol{\mu}_x. \tag{5.25}$$

respectively. Finally, the normalisation constant K_x is given by

$$K_x = \frac{\sqrt{(2\pi)^{DT}|\boldsymbol{P}_x|}}{\sqrt{(2\pi)^{3DT}|\boldsymbol{\Sigma}_x|}} \exp\Big\{ -\frac{1}{2}\big(\boldsymbol{\mu}_x^\top \boldsymbol{\Sigma}_x^{-1} \boldsymbol{\mu}_x - \boldsymbol{r}_x^\top \boldsymbol{P}_x \boldsymbol{r}_x\big)\Big\}. \tag{5.26}$$

where D is the dimension of the static feature vectors $\boldsymbol{C}$.

Omitting the normalisation constant K_x in Equation (5.21), the new trajectory HMM $\bar{\mathcal{M}}$ is formed. The probability of the static feature sequence $\boldsymbol{C}$ given the new model $\bar{\mathcal{M}}$ is

$$p(\boldsymbol{C}|\bar{\mathcal{M}}) = \sum_x p(\boldsymbol{C}|x, \bar{\mathcal{M}}) p(x|\bar{\mathcal{M}}), \tag{5.27}$$

where

$$p(\boldsymbol{C}|x, \bar{\mathcal{M}}) = \mathcal{N}(\boldsymbol{C}|\bar{\boldsymbol{C}}_x, \boldsymbol{P}_x). \tag{5.28}$$

There are a number of differences between the standard HMM $\mathcal{M}$ and the newly derived trajectory HMM $\bar{\mathcal{M}}$. Firstly, the mean sequence $\bar{\boldsymbol{C}}_x$ of the trajectory HMM given by Equation (5.22) is exactly the same as the speech parameter trajectory generated by using the SPG technique (see Equation 5.15). This indicates that the mean vector sequence of the new trajectory model may vary within in a state duration, which is constrained by the relationship between static and dynamic parameters. As a result, the state output probability of observing the static part of the output vector changes during a state occupancy, and is affected by statistics of neighboring states. Therefore, the trajectory HMM can alleviate two shortcomings of the standard HMM: constant statistics within an HMM state duration and the independence assumption of observations given the state sequence.

Secondly, in the trajectory HMM, the spectral parameter vector sequence $\boldsymbol{C}$ is modelled by a mixture of Gaussians whose dimensionality is DT, since, in general, the product of mixtures of Gaussians is a (possibly rather large) mixture of Gaussians. Thirdly, while in HMMs the covariance matrix is assumed to be diagonal, the covariances $\boldsymbol{P}_x$ of the newly derived trajectory HMM are generally full. This provides a means of capturing the inter-frame dependencies introduced by the use of dynamic features, though at the cost of increased parameterisation and computational complexity.

5.3.2 Relationships between the two techniques

The previous sections discuss the SPG technique and the trajectory HMM method proposed by Tokuda et al. Given an HMM and a state sequence, the SPG algorithm aims to solve the problem of minimising the mean square error between the training data $\boldsymbol{C}$ and the generated speech parameter trajectory $\hat{\boldsymbol{C}}$. On the other hand, the trajectory HMM defines an alternative PDF $\mathcal{N}(\boldsymbol{C}|\bar{\boldsymbol{C}}_x, \boldsymbol{P}_x)$, which is a function of the static parameters, rather than the augmented parameters $\boldsymbol{O}$. This new PDF is obtained by normalising the probability function $\mathcal{N}(\boldsymbol{W}\boldsymbol{C}|\boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x)$ given by the standard HMM

$$\mathcal{N}(\boldsymbol{C}|\bar{\boldsymbol{C}}_x, \boldsymbol{P}_x) = \frac{1}{K_x}\mathcal{N}(\boldsymbol{W}\boldsymbol{C}|\boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x), \tag{5.29}$$

where the nomalisation constant is chosen to give a valid PDF

$$K_x = \int_{\mathbb{R}^{DT}} \mathcal{N}(\boldsymbol{W}\boldsymbol{C}|\boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x)\mathrm{d}\boldsymbol{C}, \tag{5.30}$$

such that

$$\int_{\mathbb{R}^{DT}} \mathcal{N}(\boldsymbol{C}|\bar{\boldsymbol{C}}_x, \boldsymbol{P}_x) = 1. \tag{5.31}$$

The new mean trajectory $\bar{\boldsymbol{C}}_x$ is exactly the same as the speech parameter trajectory $\hat{\boldsymbol{C}}_x$ obtained by the SPG algorithm.

In both the SPG algorithm and the trajectory HMM algorithm, the mapping $\boldsymbol{W}$ between static and dynamic features plays an important role. Without this constraint, the SPG algorithm simply produces a sequence of piecewise constant mean vectors. On the other hand, given the mapping $\boldsymbol{W}$ and a sequence of static parameters $\boldsymbol{C}$, the corresponding augmented feature vectors $\boldsymbol{O}$ is completely determined. In other words, the mapping $\boldsymbol{W}$ contains all the information about the dynamic features (Δ and Δ^2). For this reason, it is possible to integrate out the dynamic features while still preserving the 'dynamic' information as long as the mapping $\boldsymbol{W}$ is retained in the trajectory HMM.

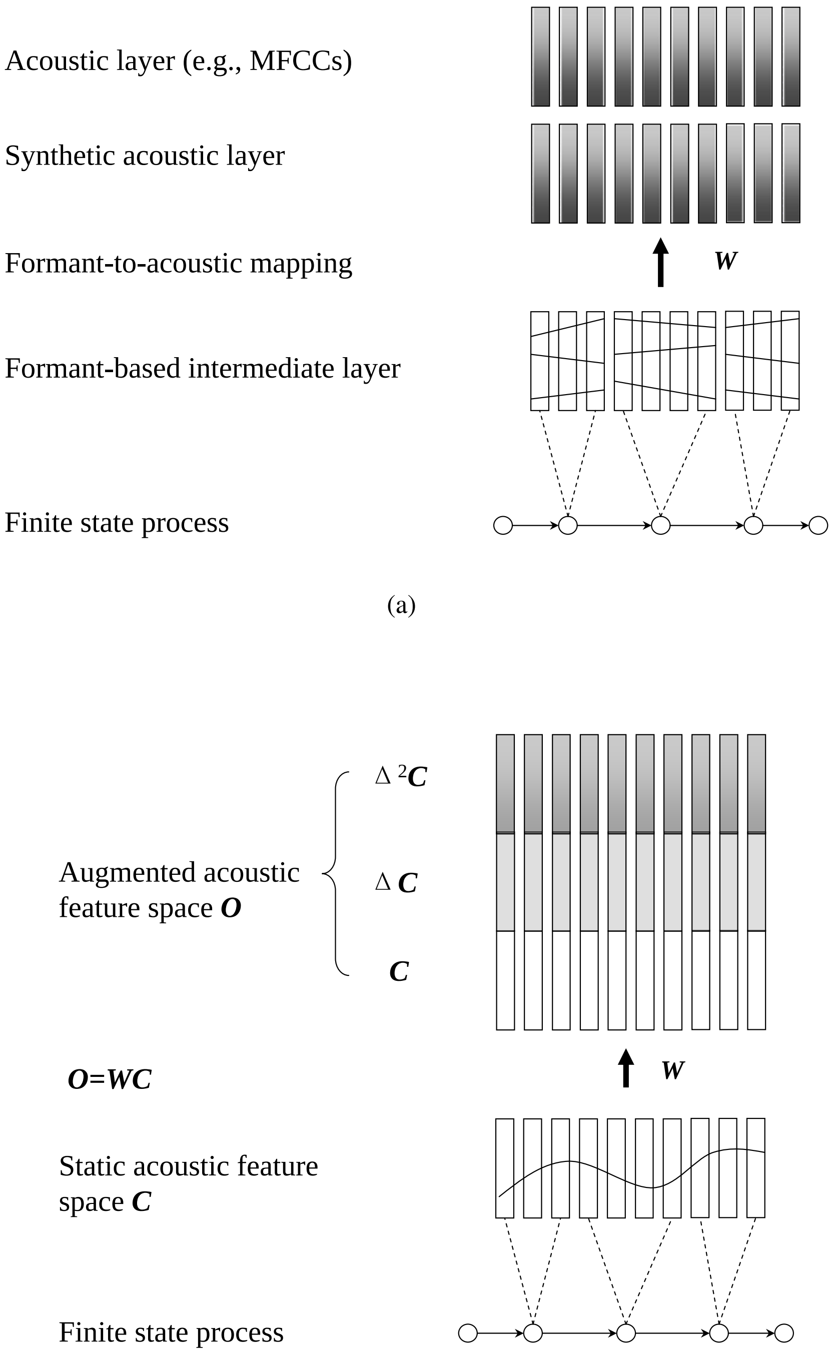
5.3.3 Comparisons between the trajectory HMM and the MSHMM

Chapter 4 describes the MSHMM and this chapter (Section 5.3) discusses the trajectory HMM. This section attempts to investigate the relationships between an MSHMM and a trajectory HMM because, intuitively, there are many similarities between these two approaches. The MSHMM and the trajectory HMM are both shown in Figure 5.1 for comparison, and Table 5.1 lists and compares corresponding elements of the two models.

	MSHMM	Trajectory-HMM
State process	Segmental HMM	Standard HMM
'Intermediate' space	Formant	Static features ($\boldsymbol{C}$)
Observed space	Static features ($\boldsymbol{C}$)	Augment features ($\boldsymbol{O}$)
Mapping(s)	$\boldsymbol{W}_k$, phone-class-dependent Linear, least-squares	$\boldsymbol{W}$, single fixed Linear, deterministic
Trajectory form	Piecewise linear	Smoothed nonlinear

Table 5.1: Comparison between a linear/linear MSHMM and a trajectory HMM.

The MSHMM is based on the segmental HMM framework, which incorporates explicit state duration models. This offers some advantages in modelling state durations compared to the trajectory HMM, which is derived from a standard HMM. Both an MSHMM and a trajectory HMM have two different, though related, speech representations. In an MSHMM, these are the acoustic representation and the formant-based intermediate representation, while in a trajectory HMM these correspond to augmented acoustic observations and static acoustic observations. In both cases, though, a fixed, linear mapping $\boldsymbol{W}$ is used to link the two


Acoustic layer (e.g., MFCCs)
Synthetic acoustic layer
Formant-to-acoustic mapping
W
Formant-based intermediate layer
Finite state process
Δ^2C
Augmented acoustic feature space O
Δ C
C
O=WC
W
Static acoustic feature space C
Finite state process

(a)

(b)

Figure 5.1: Comparison between a linear/linear MSHMM and a trajectory HMM. (a) A linear/linear MSHMM. (b) A trajectory HMM. The piecewise linear trajectories in the formant-based intermediate layer in (a) represent formant dynamics. The continuous trajectory in the static acoustic feature space in (b) represents the mean of the observation vector sequence (note that only a single dimension is shown here for simplicity).

speech representations. However, in the trajectory HMM, this mapping $\boldsymbol{W}$ is a $3DT \times DT$ matrix which is dependent on the length T of the state sequence and the dimension D of the static acoustic observation vector. In MSHMMs, the formant-to-acoustic mapping $\boldsymbol{W}$ is an $N \times M$ matrix (where N is the dimension of the acoustic space and M denotes the dimension of the formant space), which is estimated based on the least-squares technique and is also phone-class-dependent.

Both approaches have model trajectory parameters in the 'intermediate' layer (if we can think of the static feature space in the trajectory HMM as a special type of 'intermediate' layer). The ML estimation of the parameters in both models can be obtained by using a Viterbi-type training algorithm based on the EM technique. (Zen et al. (2007) gave a detailed account of the training algorithm for the trajectory HMM.) The main difference is, of course, that piecewise linear trajectories are used in a linear/linear MSHMM while the parameter trajectory in the trajectory HMM is smooth and non-linear. Obviously, the non-linear trajectories are more realistic in describing feature dynamics or formant transitions than piecewise linear trajectories. This is a major advantage of the trajectory HMM method, though at the cost of increased computational complexity. Another drawback of the trajectory HMM approach is the lack of an appropriate decoder. To date, recognition work using the trajectory HMM have almost universally resorted to the N-best list rescoring paradigm.

5.4 HMM-based speech synthesis

5.4.1 Overview of HMM-based speech synthesis

HMM-based speech synthesis (see, e.g., Yoshimura (2002)), is an alternative speech synthesis approach which has grown in popularity over the last decade. Performance of HMM-based speech synthesis continues to improve due to the emergence of new algorithms, such as the SPG algorithm (Tokuda, Kobayashi & Imai 1995, Tokuda et al. 2000) and the trajectory HMM technique (Tokuda et al. 2003, Zen et al. 2007), as discussed in Sections 5.2 and 5.3, respectively. Table 5.2 shows speech synthesis evaluation results for English comparing two

	Year							
	2007		2008		2009		2010	
System	MOS	WER	MOS	WER	MOS	WER	MOS	WER
Natural	4.7	-	4.8	22	4.9	14	4.8	12
Festival	3.0	25	3.3	35	2.9	25	3.0	23
HTS 2005	-	-	2.9	33	2.7	23	2.5	18

Table 5.2: Comparing the results of the benchmark systems for English from 2007 to 2010 in the Blizzard Challenge. 'MOS' means mean naturalness score and WER means word error rate in percent using semantically unpredictable sentences. Data taken from (King & Karaiskos 2010).

different synthesis approaches to natural speech in the Blizzard Challenge project[1] from 2007 to 2010. HTS (HMM-based speech synthesis system) consists of a class of open-source software tools for research in HMM-based speech synthesis,[2] and the University of Edinburgh's Festival Speech Synthesis System (see, e.g., Clark, Richmond, Strom & King (2006)) is one of the most successful waveform concatenation systems based on unit selection, which provides state-of-the-art synthesis performance. The listening test results presented in Table 5.2 show that the overall naturalness of the HMM-based synthesis (HTS 2005) can be comparable to the concatenative synthesis system (Festival), though both are significantly lower than the natural speech. As far as the intelligibility of the synthetic speech is concerned, the HMM-based approach has been consistently producing more intelligible (with lower WERs) synthetic speech than the Festival-based unit selection system.

A typical HMM-based speech synthesis system was described in (Tokuda, Zen & Black 2002). Generally speaking, HMM-based synthesis has two main stages: training and synthesis. First, CD HMMs are automatically trained on a large speech database in a similar manner to ASR. In the synthesis stage, the appropriate set of HMMs are concatenated according to the input label sequence, and a sequence of speech parameters are generated from these HMMs based on the SPG algorithm (Tokuda et al. 2000, as described in Section 5.2). These two stages are described in Section 5.4.2 and Section 5.4.3, respectively.

[1] The Blizzard Challenge is an international evaluation of corpus-based speech synthesisers on the same data open to any participant. Visit http://festvox.org/blizzard/index.html for more details.

[2] The 'T' in HTS stands for 'triple', i.e., H Triple S (HMM-based Speech Synthesis System). HTS was initially developed at the Nagoya Institute of Technology and and Tokyo Institute of Technology by K. Tokuda, H. Zen, T. Toda, J. Yamagishi, A. Black and T. Nose. Available online: http://hts.sp.nitech.ac.jp/.

5.4.2 Training

Training in HMM-based speech synthesis is very similar to that in speech recognition. In both cases, the standard ML criterion is typically used to estimate the model parameters for a set of context-dependent phone-level HMMs on a large amount of speech data,

$$\hat{\Lambda} = \arg\max_{\Lambda} p(\boldsymbol{O}|\mathcal{W}, \Lambda), \tag{5.32}$$

where Λ, $\boldsymbol{O}$ and $\mathcal{W}$ denote the model parameters, training data and reference transcripts, respectively. As in speech recognition, the resulting speaker-independent models are often adapted to a particular speaker in HMM-based speech synthesis so that a variety of voice characteristics may be obtained. In fact, voice adaptation is one of the major advantages of HMM synthesis over traditional concatenative synthesis. Although there are many similarities in speech recognition and HMM-based synthesis training, the following issues require extra attention in HMM-based speech synthesis.

Duration modelling

In standard HMMs, the state duration is implicitly given by a geometric model, which is inappropriate for modelling state duration. Alternative duration distributions have been investigated in the past to provide a more realistic state duration model for ASR (see Section 2.2.2 on page 19 for a variety of explicit duration models). Most HMM-based speech synthesis systems also employ explicit duration distributions to improve the duration modelling. In the HMM-based speech synthesis system described in (Yoshimura, Tokuda, Masuko, Kobayashi & Kitamura 1998, Yoshimura et al. 1999), for example, state duration for each HMM was modelled by a single multivariate Gaussian distribution, whose dimensionality equals the number of states of the HMM. The nth dimension of the state duration density represents the nth state of the HMM (the HMM is assumed to be a left-to-right model with no skip). The use of a continuous duration distribution like Gaussian allows the speaking rate of the synthetic speech to change easily.

Spectrum and pitch modelling

In HMM-based speech synthesis, not only spectral parameters (such as mel-cepstral coefficients and their time derivatives) but also excitation parameters (e.g., $\log F_0$ and its delta and delta-delta coefficients) are encoded in the HMM. The feature vector $\boldsymbol{o}_t$ therefore consists of both spectral parameters $\boldsymbol{c}_t$ and excitation parameters e_t, augmented with their respective dynamic features. The spectral vector $\boldsymbol{c}_t$ are obtained by using a mel-cepstral analysis technique described in (Fukada, Tokuda, Kobayashi & Imai 1992), which allows speech to be re-synthesised from these mel-cepstral coefficients using the mel log spectrum approximation (MLSA) filter (Imai 1983).

The spectral and excitation parameters are modelled simultaneously as separate data streams using CD phone-level HMMs. It is straightforward that the mel-cepstral coefficients can be modelled using continuous density HMMs. However, the modelling of the excitation parameter F_0 requires special treatment because for unvoiced speech sounds, F_0 is usually assumed to be undefined. As a result, the F_0 observation data combines one-dimensional continuous values for voiced regions and a discrete symbol representing 'unvoiced'. In this case, a simple continuous distribution is inappropriate to model F_0. Tokuda, Masuko, Miyazaki & Kobayashi (1999) suggested a multi-space probability distribution (MSD) HMM to deal with the F_0 pattern modelling in HMM-based speech synthesis. One option is to use a 1-dimensional Gaussian PDF to characterise F_0 patterns from the 'voiced' space and a discrete mass function for unvoiced space. In other words, the state output distribution in the MSD-HMM for F_0 is a combination of continuous and discrete distributions.

Contextual modelling

In speech recognition systems, context-dependent models, for example triphones, are commonly used to deal with contextual effects such as coarticulation. In HMM-based speech synthesis, the notion of 'context' has a much broader meaning. Both phonetic and prosodic contexts (such as rhythm, intonation and stress) are taken into account, resulting a set of full context models. Due to the increased number of contextual factors, a stream-dependent decision-tree based context clustering technique was used (see, e.g., Yoshimura et al. (1999)) to enable robust estimation of model parameters. Such an approach allows the distributions for spectral parameters, pitch parameters and the state duration to be clustered independently because the spectrum, pitch and duration each have their own influential contextual factors.

5.4.3 Synthesis

In the synthesis part, a text analyser converts an arbitrarily given text to a context-based label sequence $\mathcal{W}$. Based on this label sequence $\mathcal{W}$, a composite HMM Λ is constructed by concatenating the appropriate set of CD HMMs. Given the label sequence $\mathcal{W}$ and the composite HMM Λ, the speech synthesis problem is to generate a sequence of speech parameters $\hat{\boldsymbol{O}}$, such that

$$\hat{\boldsymbol{O}} = \arg\max_{\boldsymbol{O}} p(\boldsymbol{O}|\mathcal{W},\Lambda) = \arg\max_{\boldsymbol{O}} \sum_{\text{all}\,\mathrm{x}} p(\boldsymbol{O},\mathrm{x}|\mathcal{W},\Lambda), \tag{5.33}$$

where the sum is over all possible state sequence x. The above equation can be approximated by

$$\hat{\boldsymbol{O}} \approx \arg\max_{\boldsymbol{O}} \max_{\mathrm{x}} p(\boldsymbol{O}|\mathrm{x},\Lambda)p(\mathrm{x}|\mathcal{W},\Lambda), \tag{5.34}$$

or,

$$\hat{\mathrm{x}} = \arg\max_{\mathrm{x}} p(\mathrm{x}|\mathcal{W},\Lambda), \tag{5.35}$$

$$\hat{\boldsymbol{O}} = \arg\max_{\boldsymbol{O}} p(\boldsymbol{O}|\hat{\mathrm{x}},\Lambda). \tag{5.36}$$

The probability of $p(\mathrm{x}|\mathcal{W},\Lambda)$ can be written as

$$p(\mathrm{x}|\mathcal{W},\Lambda) = \prod_{j=1}^{N} d_j(\tau_j), \tag{5.37}$$

where N is the number of states of the HMM Λ, d_j the state duration distribution of state j and τ_j the state duration. In other words, to solve Equation (5.36), it is equivalent to determine the ‘optimal’ state durations of the composite HMM Λ. In case d_j is Gaussian, i.e., $d_j(\tau_j) = \mathcal{N}(\tau_j|\mu_j, \sigma_j)$, the probability $p(\mathrm{x}|\mathcal{W}, \Lambda)$ is maximised when $\tau_j = \mu_j$, $\forall j \in \{1, \ldots, N\}$. That is to say, the optimal state sequence $\hat{\mathrm{x}}$ is the one in which the state duration for each state takes the value of the mean for that state.

Once the state sequence is determined, the speech parameters (including both mel-cepstral coefficients and pitch parameters) can be obtained by using the SPG algorithm (Tokuda et al. 2000), which is described in Section 5.2. Finally, speech is synthesised by passing the generated mel-cepstral coefficients and pitch parameters through the MLSA filter.

5.4.4 Voice characteristics conversion

One of the major advantages of HMM-based speech synthesis is that diverse voices can be produced easily using speaker adaptation techniques (Masuko, Tokuda, Kobayashi & Imai 1997, Tamura, Masuko, Tokuda & Kobayashi 1998). A recent example of applying adaptation techniques to HMM synthesis can be found in (Yamagishi et al. 2009). Yamagishi pioneered the use of ‘average voice models’ (Yamagishi, Ogata, Nakano, Isogai & Kobayashi 2006) for HMM-based speech synthesis and investigated various model adaptation approaches to speech synthesis which can be used to generate a diversity of voices, emotion and styles (Yamagishi, Kobayashi, Tachibana, Ogata & Nakano 2007).

5.5 Summary

This chapter describes a number of techniques proposed by Tokuda and colleagues. The non-linear formant trajectories used in this study to model formant dynamics are generated by using the SPG algorithm, which is described in Section 5.2. A prerequisite of this algorithm is that both static and dynamic features must be included in HMMs. Given a particular state sequence, the SPG algorithm can be used to ‘smooth’ the sequence of piecewise constant mean vectors, resulting in a non-linear trajectory in the static feature space. In fact, it is the constraints between static and dynamic features that ensure smooth transitions between neighbouring frames and states in the generated trajectory. Without the constraint, the generated speech parameters are just a sequence of piecewise constant mean vectors.

The trajectory HMM is a recent variant of a standard HMM. Under this new trajectory model, an alternative PDF is defined on the set of static acoustic observations, whose mean vector is constrained by the relationship between static and dynamic features. The mean vector sequence is identical to the speech parameter sequence generated by using the SPG algorithm. That is to say, the mean vector changes within a state occupancy, and is also constrained at the state boundaries. The relationships between the trajectory HMM and the SPG algorithm, the standard HMM and the MSHMM are investigated.

The SPG algorithm was initially proposed to build HMM-based speech synthesisers, which generate speech directly from HMMs themselves. Many techniques developed in ASR can be readily 'shared' in HMM-based speech synthesis. Training involves simultaneously modelling of spectral, excitation and durational parameters. In particular, the MSD-HMM can be used to model pitch patterns, which employs separate distributions to model voiced (continuous distributions) and unvoiced regions (discrete distributions). Contextual factors are built into the models, which are extended from phonetic only, as in recognition, to include a great diversity of phonetic and prosodic contexts, leading to a set of 'full-context' HMMs. The SPG algorithm is used to generate a smoothed speech parameter sequence.

Both formant synthesis (see Section 3.2.4) and HMM-based synthesis are based on the vocoding technique (albeit with different vocoders) so that the the quality of the synthetic speech is, in general, not as natural as those generated by concatenative synthesisers (which are based on actual human speech samples). However, both formant synthesis and HMM synthesis require no use of the large database of actual speech samples in run-time as in concatenative synthesis systems. Therefore, the resulting systems are relatively small, which may be more attractive in situations where memory and computational power are especially limited, such as in embedded systems. In addition, formant synthesis and HMM-based synthesis can generate synthetic speech which is mostly reliably intelligible and rarely exhibits the acoustic glitches that commonly plague concatenative systems.

Chapter 6

Synthesis using non-linear trajectories

6.1 Introduction

The premise of this research is that the use of a non-linear trajectory model of formant dynamics will lead to improvements in performance for both speech synthesis and recognition, relative to a piecewise linear trajectory model. This chapter reports experiments and results on speech synthesis using a non-linear trajectory model of dynamics, and speech recognition experiments and results based on a non-linear trajectory model of formant dynamics are presented in Chapter 7. In either case, the non-linear formant trajectories are generated by using the SPG algorithm described in Chapter 5. It should be noted that the emphasis of the synthesis work is to study and evaluate the non-linear trajectory model of dynamics from the perspective of speech synthesis, rather than to develop a state-of-the-art speech synthesiser.

Both formant-based speech synthesis and HMM-based speech synthesis have been discussed in this thesis. The synthesis work presented in this chapter can be thought of as lying somewhere in between these two approaches. A basic HMM-based speech synthesis system is built, in which the HMMs are trained on 12 PFS control parameters, augmented with their dynamic features. This (excluding dynamic features) is essentially the intermediate representation of an MSHMM. A distinct feature of this work is that the speech parameter generation algorithm is applied to the 12 PFS data and followed by formant synthesis. In the context of an MSHMM, this study takes a first step towards the goal of a unified model for speech recognition and synthesis.

The rest of this chapter is organised as follows. Section 6.2 gives a detailed account of the speech synthesis system developed for this research and measures used for evaluation. Experimental results and analysis are provided in Section 6.3.

6.2 Method

The flow chart shown in Figure 6.1 outlines the main components and steps involved in the text-to-speech (TTS) conversion. Orthographic text is converted into a sequence of phonetic segments using a pronunciation dictionary. Each phonetic segment in the phonetic segment sequence is associated with a corresponding HMM in the HMM store, which is a collection of standard HMMs built on 12 PFS control parameters and their dynamic features. Two types of formant synthesiser control parameter sequences are generated, namely piece-wise constant PFS control parameters and non-linear, smoothed PFS control parameters. The latter is generated by using the SPG algorithm. The HMS parallel formant synthesiser is used to convert the 12 PFS control parameter sequence into speech.

6.2.1 Data

The TIMIT training set was divided into two parts based on the gender of the speakers so that synthesis models for both male and female speakers can be built. The training set for male speakers consists of 326 male speakers and the female set comprises 136 female speakers (10 sentences spoken by each speaker). Each waveform file was downsampled to 8KHz and analysed by the Holmes formant analysis toolkit to generate a sequence of 12-dimensional PFS control parameters.[1] The resulting 12 PFS control parameters were then converted to the HTK format to be compatible with HTK.[2] The application of the SPG algorithm (see Section 5.2) requires both static and dynamic feature vectors. Therefore, the original 12 PFS control parameters were augmented with Δ and Δ^2 coefficients, based on Equations (5.1, 5.2) using HTK, resulting in a sequence of 36-dimensional feature vectors.

[1] The Holmes formant analyser and the calculation of the 12 PFS control parameters are described in detail in Section 3.2.3 on page 59.

[2] Thanks to Martin Russell for providing the source code for PFS-to-HTK conversion.

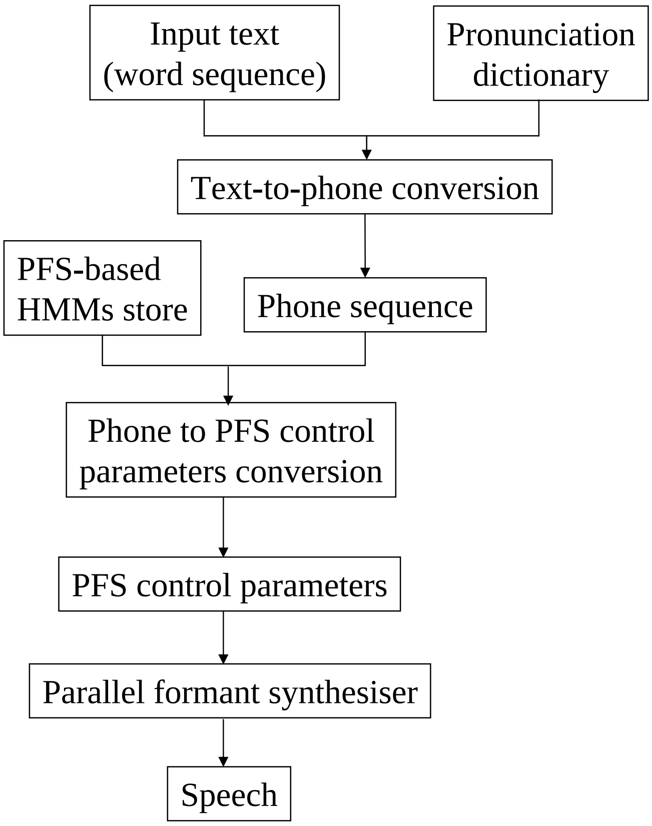


Figure 6.1: Flow chart showing the main steps involved in the text-to-speech conversion.

6.2.2 Text-to-phone conversion

The TTS system can take any sequences of phonetic segments as input, though for word input a conversion is needed in the first place. A text-to-phone converter was built to transform a word string into a sequence of context-sensitive phonetic labels using the TIMIT phonetic lexicon. This dictionary contains all English words (a total of 6,232 words) that appear in the TIMIT text material. The pronunciations contained in the TIMIT lexicon are based on the original 64-phone set. These pronunciations are modified to be compatible with the 49-phone set (see appendix A), as used in previous work (Russell & Jackson 2005).

The size of the TIMIT dictionary is fairly small for a practical speech synthesis system. However, as stated before, the goal is to evaluate the non-linear trajectory model by means of speech synthesis, rather than to develop a state-of-the-art speech synthesiser. From this perspective, the TIMIT dictionary is adequate for this research.

6.2.3 Model sets

A variety of models were built for speech synthesis experiments. These models are gender-dependent so that they can be divided into two broad categories: models for male speakers and models for female speakers. Within either category, the baseline is a set of speaker-independent (SI) phone-level CI 3-state left-to-right single Gaussian HMMs. Complexity were then gradually added to develop more sophisticated models. Briefly speaking, the following three options have been investigated. Firstly, triphone CD models were built to improve the contextual modelling in the HMMs. Secondly, the 3-state single Gaussian topology of HMMs was extended to include more states per phone (ranging from 3 to 6-state per phone). Thirdly, the SI models were adapted to a particular speaker to build speaker-dependent (SD) models. All these models were built and trained using HTK.

Baseline models

A set of 49 3-state left-to-right (with no skips) single Gaussian CI HMMs, including a model for silence ('sil'), was built based on the TIMIT 49-phone set. Firstly, 49 identical HMMs was created, in which all of the Gaussian densities were set to have zero means with unit variances. These 49 models were then initialized using the label files provided with TIMIT. Next, the model parameters were optimized using the standard forward-backward algorithm (Dempster et al. 1977, Liporace 1982), which is implemented in the HTK tool **`HERest`**. For each training utterance, a composite HMM is created based on the corresponding label file in such a way that the individual phone HMMs corresponding to each label in the transcript are concatenated together to form a single, composite model. Then the forward and backward probabilities for the composite HMM are calculated. Given the forward and backward probabilities, the probabilities of state occupation at each time frame are computed, and the sums needed to form the weighted averages are accumulated in the normal way of HTK. The model parameters will not be updated until all training utterances have been processed. Then, the new parameter estimates are computed from the weighted sums for all HMMs in the model set.

Context-dependent models

Context-dependent triphone HMMs were constructed based on the 49 CI models. The use CD models led to a significant increase in the number of models and parameters. In order to enable robust model parameters estimation, similar states were tied using a phonetic decision tree (Young, Odell & Woodland 1994). The resulting triphone model set comprised 6575 physical models. Model parameters were re-estimated using the forward-backward algorithm, as in the case of CI model parameters estimation.

HMMs with more states

The models obtained so far are 3-state single Gaussian models. These models might not be ideal for speech synthesis and can be refined further by increasing the number of states per model or using more mixtures per state. Since mixture components can be considered as a special form of sub-state in which the transition probabilities are the mixture weights, this research opts to use more states per model to improve the HMMs. The number of states per model varies from three to six, as it was observed that models with more than six states often cause problems in parameter estimation due to insufficient training data.

Speaker-dependent models

In speech recognition, it is common practice to adapt the SI models to obtain a set of speaker-dependent (SD) models, which are trained on, and subsequently used to recognise, the speech of a single speaker. Maximum likelihood linear regression (MLLR) (Leggetter & Woodland 1995) or MAP adaptation are commonly used for speaker adaptation. Using SD models, the target speaker's characteristics can be modelled more accurately and hence the recognition performance is substantially improved. Lee & Giachin (1991) showed that error rates for SD systems are typically two to three times lower than equivalent SI systems. As far as HMM-based synthesis is concerned, model adaptation is of great interest because one can generate synthetic speech which, in principal, sound like a particular individual's voice (see, e.g., Yamagishi & Kobayashi (2007*a*), Yamagishi et al. (2009)).

Model adaptation using MLLR MLLR adaptation involves estimation of a set of linear transformation matrices, which are used to adapt the model mean vectors to maximise the likelihood of the adaptation data. The number of the transformation matrices depends on the amount of available adaptation data. When only a small amount of adaptation data is available, a global transformation may be used. Given a continuous HMM with Gaussian state output PDFs, each state s_j is fully defined by the mean vector $\boldsymbol{\mu}_j = [\mu_1, \dots, \mu_D]^\top$ and the covariance matrix $\boldsymbol{\Sigma}_j$ (D is the dimension of the acoustic observation space). Using the MLLR technique, the adapted mean $\hat{\boldsymbol{\mu}}_j$ is given by

$$\hat{\boldsymbol{\mu}}_j = \boldsymbol{W}\boldsymbol{\xi}_j, \tag{6.1}$$

where $\boldsymbol{W}$ is a $D \times (D+1)$ linear transformation matrix which maximises the likelihood of the adaptation data. $\boldsymbol{\xi}_j$ is the extended $(D+1) \times 1$ mean vector, defined as $\boldsymbol{\xi}_j = [\omega, \mu_1, \mu_2, \dots, \mu_D]^\top$, where ω is a bias offset. Leggetter & Woodland (1995) reported MLLR adaptation experiment results on the Resource Management RM1 database. It has been shown that just three adaptation utterances (equivalent to an average of 11s of speech per speaker) can result in improvement in recognition rates using a global transformation. When more adaptation data is available, the performance gradually improves and it is also possible to train multiple transforms.

Model adaptation using MAP Maximum *a posteriori* estimation provides a means of incorporating prior information of the parameters to be estimated in the model training process. Given a training observation sequence $\boldsymbol{O} = \{\boldsymbol{o}_1, \dots, \boldsymbol{o}_T\}$, the MAP estimation of the model parameters Λ_{MAP} is obtained by

$$\Lambda_{\text{MAP}} = \arg\max_{\Lambda} P(\Lambda|\boldsymbol{O}) = \arg\max_{\Lambda} p(\boldsymbol{O}|\Lambda)P(\Lambda), \tag{6.2}$$

where $P(\Lambda)$ is the prior probability of Λ. As can be seen, if $P(\Lambda) \equiv \text{constant}$, which corresponds to a non-informative prior, Equation (6.2) reduces to the standard maximum-likelihood approach. For MAP adaptation, the SI model parameters are usually used as the informative priors. Given the SI model mean $\boldsymbol{\mu}$ and the adaptation speech data mean $\bar{\boldsymbol{\mu}}$, following the HTK implementation, the state mean is updated by

$$\hat{\boldsymbol{\mu}} = \frac{\psi}{\psi + \chi}\bar{\boldsymbol{\mu}} + \frac{\chi}{\psi + \chi}\boldsymbol{\mu}, \tag{6.3}$$

where χ is a weighting of the *a priori* knowledge (i.e., the SI model parameters) to the adaptation speech data and ψ is the occupation likelihood of the adaptation data. It can be seen from Equation (6.3) that if the occupation likelihood of the adaptation data ψ is small, the estimated mean $\hat{\boldsymbol{\mu}}$ will be close to the SI model mean $\boldsymbol{\mu}$. For this reason, MAP adaptation generally requires more adaptation data than MLLR adaptation.

Both MLLR and MAP were used in the speech synthesis experiments. Speaker adaptations have been used in HMM-based synthesis in the past (Tamura et al. 1998, Yamagishi & Kobayashi 2007*a*, Yamagishi et al. 2009), though they are usually used in the spectral domain. In the synthesis experiments described here, model adaptation is directly applied to the formant domain. One of the potential advantages of adaptation in the formant domain is that the results maybe more interpretable. For example, adapting an adult acoustic model to a child's speech should results in predictable changes in formant frequencies (Russell 2004).

6.2.4 Generation of formant synthesiser control parameters

Given a set of HMMs and a phone sequence, the synthesis problem is to generate a sequence of 12 PFS control parameters from the HMMs, which is then used to drive the parallel formant synthesiser. It has been shown in Section 5.4.3 that the synthesis part in HMM-based speech synthesis can be divided into two stages: 1) to decide an 'optimal' state sequence $\hat{\mathrm{x}}$ given the model Λ and the phone sequence $\mathcal{P}$ to be synthesised, such that $\hat{\mathrm{x}} = \arg\max_{\mathrm{x}} p(\mathrm{x}|\mathcal{P}, \Lambda)$, and 2) to generate a sequence of speech parameters $\hat{\boldsymbol{O}}$ based on the optimal state sequence $\hat{\mathrm{x}}$, such that $\hat{\boldsymbol{O}} = \arg\max_{\boldsymbol{O}} p(\boldsymbol{O}|\hat{\mathrm{x}}, \Lambda)$. For 1), if the state duration for each state is known, then the state sequence can be derived. Hence, the problem is actually to determine the state duration for each state.

Two methods have been studied in this research to generate a sequence of 12 PFS control parameters from HMMs, which result in two types of formant synthesiser control parameters. In the first method, two simplifying assumptions are made in generating the synthesiser control parameter sequence. Firstly, the state sequence is decided based on the expected state duration. Suppose that the self-transition probability for state s_i of an HMM is a_{ii}, where the model corresponds to a phone in the phone sequence $\mathcal{P}$. Since the state duration in the HMM follows a geometric distribution, the expected duration l_i for state i is given by $l_i = \frac{1}{1-a_{ii}}$. The state duration for the rest of the states can be determined in the same way. Based on the expected state duration, the state sequence $\hat{\mathrm{x}}$ for the phone sequence $\mathcal{P}$ can be obtained as $\hat{\mathrm{x}} = \{s_1 \times l_1, \ldots, s_N \times l_N\}$, where $s_i \times l_i$ denotes a duration of l_i spent in state s_i, and N is the number of states in the composite model corresponding to the phone sequence.

The second simplifying assumption is that the state mean vector $\boldsymbol{\mu}_i$ is generated each time the state s_i is visited. That is to say, the synthesis control parameters generated is a sequence of state mean vectors $\hat{\boldsymbol{O}} = \{\boldsymbol{\mu}_1 \times l_1, \ldots, \boldsymbol{\mu}_N \times l_N\}$, where $\boldsymbol{\mu}_i \times l_i$ denotes a l_i-length sub-sequence speech parameters with the same value $\boldsymbol{\mu}_i$ generated by state s_i (based on the piecewise stationarity assumption of HMMs). For these reasons, the resulting synthesiser control parameters using the first method are referred to as *piecewise constant* synthesiser control parameters in this thesis.

The second type of synthesiser control parameters is generated based on the same state sequence obtained in the first approach, i.e., $\hat{\mathrm{x}} = \{s_1 \times l_1, \ldots, s_N \times l_N\}$. However, this time the SPG algorithm (as described in Section 5.2, case 1) is applied to generate a non-linear, smooth trajectory of PFS control parameters given the state sequence $\hat{\mathrm{x}}$. Speech synthesiser control parameters generated using this approach are referred to as *non-linear trajectory* PFS control parameters.

Of course, the duration model in HMMs is poor, and the state sequence derived using the above approach is certainly not ideal for speech synthesis. State-of-the-art HMM-based synthesisers usually employ explicit duration distributions, as discussed in Section 5.4.2, to improve duration modelling. However, the objective of this research is not to build a state-of-the-art HMM synthesiser but to build a non-linear trajectory model, and synthesis is chosen as another way to assess this non-linear trajectory model. Although the state sequence obtained by using the approach described above is less realistic, it provides a baseline for the development of a non-linear trajectory model, and it is the behavior and performance of this non-linear trajectory model that we are interested in this research, rather than maximizing the quality of the synthetic speech.

6.2.5 Synthetic speech evaluation

Intelligibility and naturalness are two important factors when evaluating speech synthesis performance. Naturalness describes how closely the synthetic speech sounds like human speech, while intelligibility is the ease with which the output is understood (He 2001). Since model adaptation is used in this research, similarity is an important indicator which determines the effectiveness of the adaptation. Therefore, intelligibility, naturalness and similarity are the three criteria used in this research to evaluate the synthetic speech.

Subjective measures are most commonly used in synthetic speech evaluation. After all, there is no better way to evaluate a synthetic speech than to actually listen to it. However, subjective tests are typically expensive to run and results are usually not reproducible. Objective tests are often used to overcome these disadvantages of subjective tests. The results of objective tests on their own may not be as valid as subjective test results, but they can be very useful to supplement subjective tests. Both subjective and objective measures are employed in this study. For subjective tests, a total of 20 native English speakers participated in the subjective experiments.

Intelligibility test

Only subjective measures were used in the intelligibility test since objective measures are usually designed to predict speech quality rather than intelligibility. The Diagnostic Rhyme Test (DRT) (Voiers 1977, Pratt 1984) is chosen to evaluate the intelligibility of the synthetic speech. DRT is a word level speech intelligibility test which is designed to test

Attribute	Word-pair examples
Voicing	veal - feel
Nasality	meat - beat
Sustension	vee - bee
Sibilation	zee - thee
Graveness	weed - reed
Compactness	yield - wield

Table 6.1: Example word-pairs used in the DRT test.

the intelligibility of consonants which are at initial positions of words. There are totally 96 word-pairs in DRT, which are divided into 6 groups to test 6 different acoustic features (or attributes) in the initial consonant, i.e. *Voicing, Nasality, Sustension, Sibilation, Graveness and Compactness*, as shown in Table 6.1. The full 96 word-pairs are taken from (Pratt 1984), which are listed in Appendix B.

Each participant was provided with the same 96 word-pairs list. A recording was prepared beforehand, which contained a string of 96 synthetic words, each randomly chosen from the corresponding word-pair in the list. Each time a word was played, the participant chose a word from the corresponding word-pair that he/she thought was correct. Intelligibility is measured as the number of correct answers divided by the number of all words (96). Each subject was asked to do the test twice, one based on piecewise constant synthesiser control parameters and the other based on non-linear trajectory parameters, though the subject had no idea which test was based on piecewise constant control parameters or non-linear trajectory parameters. The models used to generate these synthesiser control parameters are baseline models for male speakers (see Section 6.2.3).

Naturalness test

Both subjective and objective measures are used in the naturalness test. In the subjective naturalness test, each subject hears 20 synthetic utterances, in which 10 utterances are based on piecewise constant synthesiser control parameters and the other 10 are based on non-linear trajectory synthesiser control parameters. Baseline models for male speakers are used to generate these synthesiser control parameters. The listener is asked to give their subjective impression on the overall quality of the speech using the mean opinion score (MOS) method, i.e., '**Excellent/Good/Fair/Poor/Bad**'. Subjects are prompted to take account of factors such as naturalness, listening effort, speaking style, comprehension problems, pronunciation, speaking rate and voice pleasantness at the beginning of the test. Each subject hears a different set of 20 synthetic utterances.

As for objective measures, Perceptual Evaluation of Speech Quality (PESQ) is used to assess the quality of the synthetic speech. PESQ is an objective method for end-to-end speech quality assessment of narrow-band and wide-band telephone networks and speech codecs (Rix, Beerends, Hollier & Hekstra 2001). PESQ was developed by KPN Research, the Netherlands and British Telecommunications (BT), and was officially approved as International Telecommunication Union (ITU)-T recommendation P.862.[3] In a typical PESQ test, a reference speech signal and a degraded speech signal are both transformed into psychophysical representations which approximate human perception. Their perceptual distance is calculated and mapped to give a PESQ-MOS score, which can be transformed to produce a MOS-LQO (listening quality objective, an estimation of subjective listening quality using an objective measurement technique) score. The PESQ software used in this research is the reference implementation of ITU-T P.862, P.862.1 and P.862.2, enclosed in ITU-T P.862 Annex A (2005).

In the naturalness test, PESQ is employed to measure quality for synthetic speech used in the subjective experiments such that the correlation between the subjective results and PESQ results can be analyzed. In addition, PESQ is used to evaluate synthetic speech which are generated from many different types of models, including male/female models, CI/CD models and models with different number of states. In this case, using a subjective measure would be impractical because of the large number of situations. For PESQ experiments on synthetic speech generated from gender-dependent models, 128 utterances from 16 male speakers in the TIMIT core test set and 128 utterances (64 from the core test set and 64 randomly chosen) from 16 female speakers in the TIMIT test set are chosen as the reference speech. The final PESQ score is the average of the 128 synthetic utterances.

[3] Visit http://www.itu.int/rec/T-REC-P.862/en for more details on the ITU-T P.862 recommendation.

Mimicry test

The use of MLLR and MAP techniques for speaker adaptation is described in Section 6.2.3. The purpose of the Mimicry test is to evaluate the performance of these techniques for speech synthesis. Specifically, the Mimicry test aims to test the perceptual similarity between an individual's original speech and the synthetic speech, which is generated from a set of models adapted to that individual. (Again, baseline models for male speakers are used in the mimicry test.) By perceptual similarity, it means to test whether the listener thinks that the two utterances are spoken by the same speaker. In this experiment, the similarity is measured on a subjective scale of $1 - 10$, where 10 means the two utterances are spoken by exactly the same speaker and 1 indicates that they are spoken by two completely different speakers.

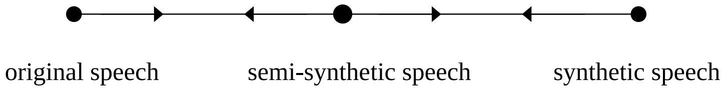


Figure 6.2: The Mimicry test using semi-synthetic speech as a 'bridge' between the original speech and the synthetic speech.

Subjective tests are conducted to evaluate the perceptual similarity. A trial experiment showed that direct comparison between original speech and synthetic speech was very difficult due to the low quality of the synthetic speech. As a compromise, 'semi-synthetic' speech is introduced as an intermediate type of speech to assist the comparison between original speech and synthetic speech, as shown in Figure 6.2. A semi-synthetic utterance is obtained in two stages. Firstly, the original speech is passed through the Holmes formant analyser to derive a sequence of 12 PFS control parameters. Secondly, these formant synthesiser control parameters are sent to the HMS parallel formant synthesiser to re-synthesise the so-called semi-synthetic speech. This process is also known as copy synthesis (Holmes 1989). As a result, the comparison of the perceptual similarity is made firstly between an original utterance and a semi-synthetic utterance, and then between the semi-synthetic utterance and the

Ori - Semi	S - S
	S - D
	D - D
Semi - Syn	MLLR
	MAP

Table 6.2: A summary of the Mimicry test.

synthetic utterance.

Both MLLR and MAP adaptation techniques are used in the Mimicry test. Moreover, the number of adaptation utterances varies, so that the effect of the amount of adaptation data on the similarity perception can be monitored. In the TIMIT speech corpus, each subject speaks only 10 sentences. Therefore, the maximum number of adaptation utterances is 9 in the mimicry test. The number of adaptation utterances is chosen as 0 (without adaptation), 5, 9 and $\bar{9}$, where $\bar{9}$ indicates that the model set is adapted to a different speaker using 9 adaptation utterances from that speaker.

The Mimicry test is summarized in Table 6.2, which can be divided into two sub-tests, namely ‘Ori - Semi’ and ‘Semi - Syn’. The first sub-test, ‘Ori - Semi’, intends to evaluate the similarity based on an original utterance and a semi-synthetic utterance. It was hoped that this sub-test would give good results as a semi-synthetic speech is obtained using copy synthesis of the original speech. In this sub-test, each subject hears 10 utterance pairs. Each utterance pair contains an original utterance and a semi-synthetic utterance. There is no adaptation involved in this sub-test. The test utterance pairs are split into three categories, referred to as S-S, S-D and D-D, according to the text material of the sentences (what is being said) and the speaker identity (who is saying). The first letter (S or D) indicates the same or different speakers, and the second letter indicates the same or different utterances. The first group (S-S) includes an original utterance and a semi-synthetic utterance from the same speaker saying the same sentence. In the second group (S-D), the sentences are different but they are spoken by the same speaker (or semi-synthesised by using the same speaker’s utterance). Finally, ‘D-D’ means the utterance pair contains different sentences and they are obtained from different speakers. The 10 utterances that each subject heard is a mixture of utterances from the above three categories.

The second sub-test of the Mimicry test is referred to as ‘Semi - Syn’, which tests the perceptual similarity between semi-synthetic speech and synthetic speech. Each subject hears eight utterance pairs, each containing a semi-synthetic utterance and a synthetic utterance. The eight synthetic utterances are generated using a combination of different adaptation techniques (either MLLR or MAP) and different number of adaptation utterances (0, 5, 9 and $\bar{9}$). In this sub-test, each utterance pair uses the same sentence. The 8 utterance pairs cover all possible combination of adaptation techniques and number of adaptation utterance, including 4 utterances based on MLLR using 0, 5, 9 and $\bar{9}$ adaptation utterances and 4 utterances based on MAP using 0, 5, 9 and $\bar{9}$ adaptation utterance. Each subject hears 8 different utterance pairs, and a different set of 8 utterance pairs is provided for each subject. It is expected that the perceptual similarity would increase as the number of the adaptation utterances increases. In the case where nine adaptation utterances from another speaker are used, it is expected that there would be a significant drop in the similarity between the synthetic speech and the semi-synthetic speech.

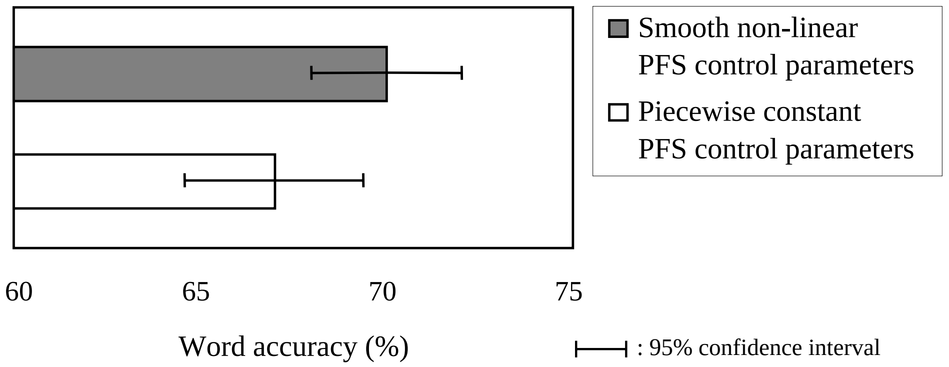
Finally, PESQ is used to assist in the mimicry test. In this case, it is interesting to find out whether the use of more adaptation utterances can lead to improved synthetic speech quality.

6.3 Synthesis evaluation experiment results and analysis

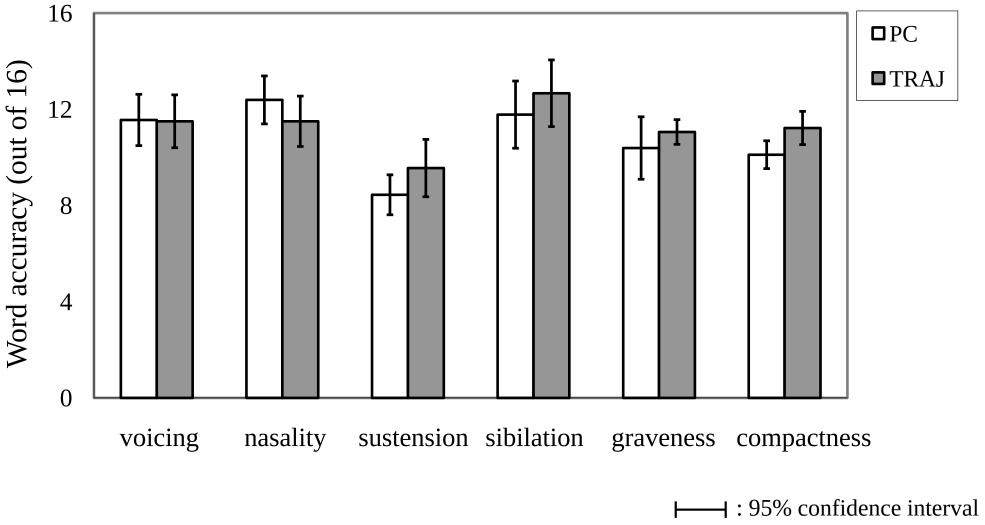
This section presents synthesis evaluation experiment results and analysis. All subjective experiment results are compiled taking account of information from all of the 20 individuals. Experiment results for DRT and Naturalness test are presented in Sections 6.3.1 and 6.3.2, respectively, with an analysis of these two experiments presented in Section 6.3.3. The Mimicry test results and analysis are presented in Sections 6.3.4 and 6.3.5, respectively.

6.3.1 DRT results

The DRT results are shown in Figure 6.3. There is no surprise that the intelligibility of the synthetic speech based on non-linear smooth trajectory formant synthesiser control parame-



(a)



(b)

Figure 6.3: (a) The DRT results for synthetic speech generated by using piecewise constant and non-linear smooth trajectory synthesiser control parameters. (b) The DRT results analyzed in terms of individual acoustic attribute, where 'PC' stands for piecewise constant synthesiser control parameters and 'TRAJ' means non-linear, smooth trajectory synthesiser control parameters.

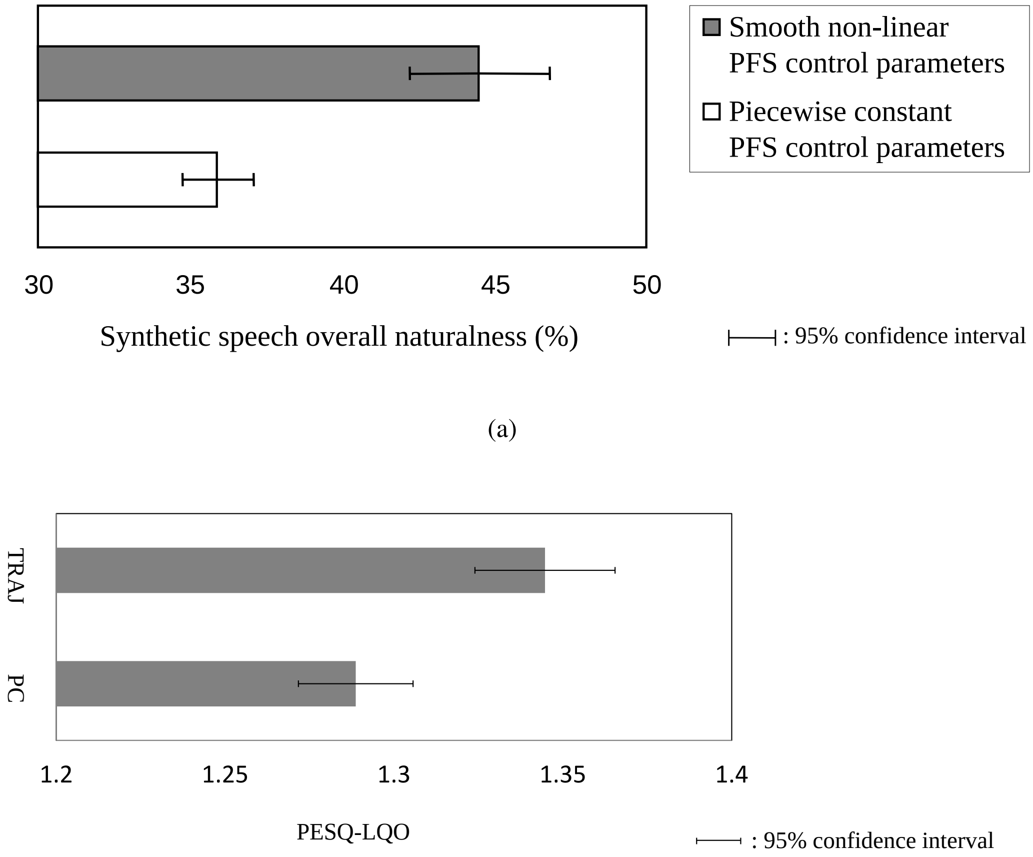
ters is higher than synthetic speech generated using piecewise constant control parameters, with word accuracies of 70% and 67% respectively, as can be seen in Figure 6.3(a). Statistical significance test results will be presented in Section 6.3.3.

In order to find out which of the 6 acoustic attribute contributes most to the overall results, a detailed analysis of the DRT results is conducted and the results are shown in Figure 6.3(b). The average scores for each individual acoustic attribute are presented and comparison is made between the two types of synthesiser control parameter sequences. Figure 6.3(b) shows that the average scores for the 'voicing' attribute are approximately the same for synthetic speech generated by both piecewise constant and non-linear trajectory synthesiser control parameters. In the case of 'nasality', however, the use of non-linear trajectory synthesiser control parameters results in a slightly lower intelligibility score, compared with speech generated by piecewise constant control parameters. These results seem to suggest that there is no benefit of using non-linear trajectory synthesiser control parameters in discriminating voiced/unvoiced and nasal/oral sounds with this particular type of speech synthesis.

The results for the rest of the 4 acoustic attributes, i.e., sustension, sibilation, graveness and compactness, follow the same trend, with non-linear trajectory control parameters giving higher scores than piecewise constant synthesiser control parameters. In addition, it can be seen from Figure 6.3(b) that among the 6 acoustic attributes, 'sibilation' gives the best score, while 'sustension' provides the lowest score for non-linear, smooth synthetic speech. For piecewise constant control parameters, 'sustension' still gives the lowest score but 'nasality' gives the highest score.

6.3.2 Naturalness test result

The results for the naturalness test are straightforward and comply with what was expected before the test. Figure 6.4(a) shows the result for subjective naturalness test. The use of non-linear trajectory synthesiser control parameters yields a higher score, 45%, while the use of piecewise constant synthesiser control parameters produces a score of 35.9%. Objective naturalness test results for the same set of synthetic speech data are presented in



(a)

(b)

Figure 6.4: (a) Subjective naturalness test results. (b) Objective naturalness test results, where ‘PC’ stands for piecewise constant synthesiser control parameters and ‘TRAJ’ means non-linear, smooth trajectory synthesiser control parameters.

Figure 6.4(b). As can be seen, the objective PESQ results confirm the subjective results as shown in Figure 6.4(a), in which synthetic speech generated using non-linear trajectory synthesiser control parameters give a higher PESQ score than synthetic speech generated using piecewise constant synthesiser control parameters.

As with the DRT experiments, the emphasis in the naturalness experiments is on the *relative* performance of the non-linear trajectory model and the piecewise constant model. Both subjective and objective naturalness test results clearly demonstrate the superiority of a non-linear trajectory model. However, as can be seen from Figure 6.4, the overall quality of the synthetic speech is poor. Of course, the direct reason would be that the formant synthesiser control parameters generated from HMMs are inappropriate because it is known that given an appropriate set of synthesiser control parameters, a parallel formant synthesiser can generate high quality synthetic speech (Holmes 1973). There are a number of reasons which might help to explain the poor overall quality of the synthetic speech.

Firstly, automatic estimation of formant frequencies are prone to errors. Errors occurring at the formant analysis stage are non-recoverable and inevitably introduce errors to the acoustic models. Inaccurate acoustic models will certainly degrade the quality of the synthetic speech. Secondly, the acoustic models used in the synthesis experiments summarized in Figure 6.4 are baseline HMMs for male speaker (3-state CI single Gaussian HMMs), while state-of-the-art HMM-based synthesis systems use a large set of CD models to account for contextual factors. Thirdly, no prosodic structure is included in the speech synthesis system. For these reasons, the overall quality of the synthetic speech is rather disappointing.

Improved models have been built for both male and female speakers, as discussed in Section 6.2.3, and these models have been used to generate synthetic speech. Figure 6.5 summarizes the PESQ results for synthetic speech generated from a variety of models, including CI and CD models, models with different number of states, and models for both male and female speakers. For each model set, two types of synthetic speech are generated by using the piecewise constant model and the non-linear trajectory model. Each data point in Figure 6.5 is the average PESQ-LQO score of 128 synthetic speech utterances for either male or female speakers (see Section 6.2.5 on page 112 for the description of this test).

A number of observations can be made based on the results presented in Figure 6.5. First, objective test results for male synthetic speech are better than female results. A likely reason is that there are more training utterances for male speakers (3260 utterances) than for female speakers (1360 utterances) in the TIMIT database. Therefore, model parameters are estimated more accurately for the male system. Second, the use of CD models can improve the synthetic speech quality, relative to the use of CI models, *provided that CD model parameters are well estimated.* Since there is less training data for female speakers, CD model parameters for female are not as well estimated as in the case of male. As a result, the performance gain when moving from CI to CD models is relatively small in the

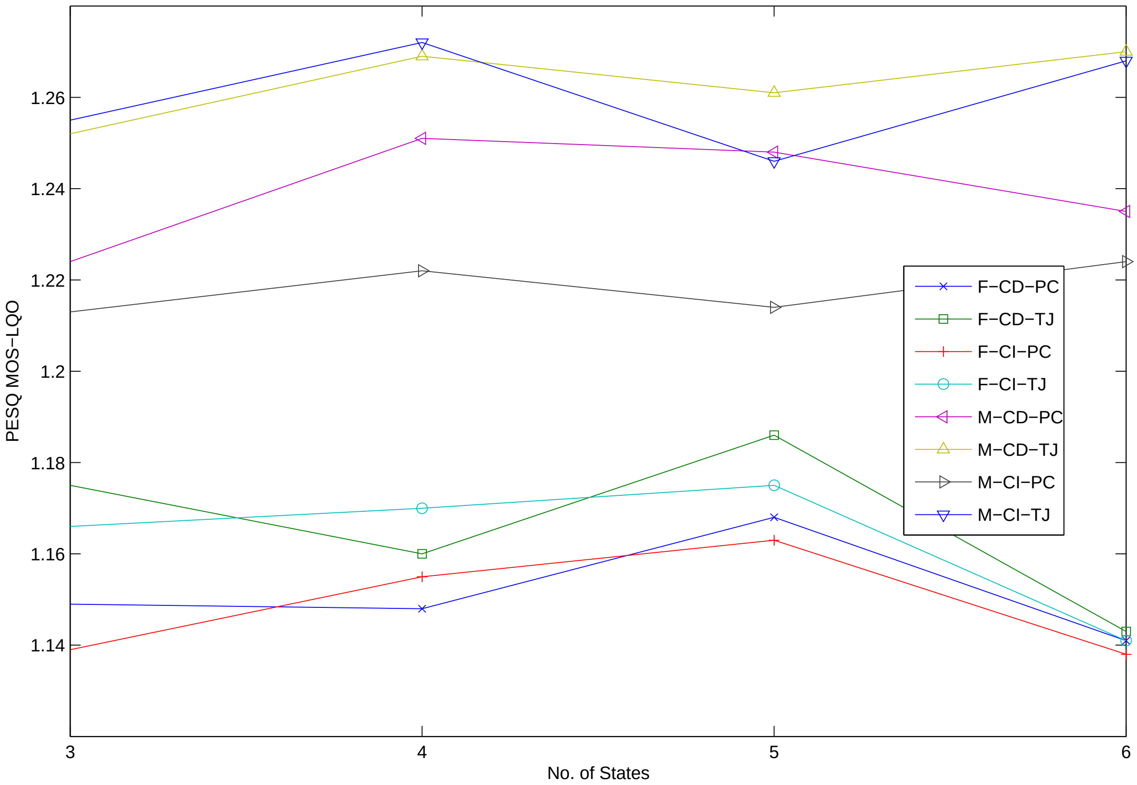


Figure 6.5: Objective naturalness test results for synthetic speech generated from various model sets. In the legend, F/M indicates female or male, and CI/CD denotes context-independent or context-dependent. PC/TJ refers to piecewise constant or non-linear trajectory synthesis control parameters.

female case (see F-CI-PC and F-CD-PC) than in the male case (see M-CI-PC and M-CD-PC), and when 4-state per phone is used, CD models for female actually degrade the quality of the synthetic speech than corresponding CI models. This can also be counted as the data-sparsity problem, which the non-linear trajectory model aims to overcome. Third, in most cases, increasing the number of states in the HMM can lead to improvement in the PESQ score, though the improvement is modest. It appears that HMMs with 4-state per phone would give the best result for the male system, while optimal performance is achived by using 5-state per HMM for the female system. Although the use of more states per HMM allows for a finer description of the underlying state process, data-sparsity problems worsen when the number of states increases because more training data is required. For example, the objective results for female system see a significant decrease in PESQ score when the number of state turns from 5 to 6, as shown in Figure 6.5, indicating that the advantage due to the use of more states per phone has been compromised by insufficient training data.

Last but not least, it is not surprising that the use of a non-linear trajectory model can improve the quality of the synthetic speech over a piecewise constant model (see M-CI-PC and M-CI-TJ, M-CD-PC and M-CD-TJ in Figure 6.5). This has already been seen in results shown Figure 6.4, where only CI models are used. When CD models are also used, as shown in Figure 6.5, it is interesting to see that a non-linear trajectory CI model can outperform a CD model with piecewise constant means (see M-CI-TJ and M-CD-PC). This result is of great importance, as it shows that the non-linear trajectory model is accounting for contextual effect such as coarticulation, which CD models are designed to account for. From the perspective of coarticulation modelling, the advantage of the non-linear trajectory approach is that it requires far few model parameters (the same as a typical CI system) than the conventional CD models. Therefore, the data-sparsity problem can be alleviated.

6.3.3 DRT and Naturalness test results analysis

First of all, a statistical significance test is used to find out whether the differences between the experimental results based on synthetic speech generated from the piecewise constant model and those generated from the non-linear trajectory model are significant or not.

Statistical significance analysis

The non-linear trajectory PFS control parameters are generated by using the SPG algorithm, which in effect performs an utterance-level smoothing on the piecewise constant synthesiser control parameters. Therefore, each observation in one sample (measures based on the piecewise constant model) is paired with a matched observation in the other sample (measures based on the non-linear trajectory model). This is a typical before-and-after scenario, for which a paired t-test is a suitable statistical significance test. For this test to be valid, the differences for the matched-pairs should be approximately normally distributed.

Therefore, for small samples, a normality test (such as a Kolmogorov-Smirnov test or a Lilliefors test) is usually performed in order to test the validity of the paired t-test before it actually takes place.

A one-tailed paired t-test is used to test the null hypothesis that the true mean of difference $\bar{d}$ between two samples is zero, i.e., $H_0 : \bar{d} = 0$, against the alternative hypothesis, $H_1 : \bar{d} > 0$, by computing

$$t = \frac{\bar{d}}{s_d/\sqrt{n}}, \tag{6.4}$$

where n is the sample size (or the number of pairs) and s_d is the standard deviation of the differences between observations of the two samples. This t-statistic is compared to a t-distribution with a $df = n - 1$ degrees of freedom to find the p-value of the paired t-test. If the p-value associated with t is low (e.g., < 0.05), there is evidence to reject the null hypothesis and accept the alternative hypothesis. Therefore, we can conclude that there is evidence that there is a significant difference in means across the paired observations.

	$\bar{d}$	s_d	df	t	p
DRT	2.9	5.2	19	2.49	0.01
Naturalness	4.55	4.29	19	4.75	$<$ 0.001

Table 6.3: t-test results for DRT and subjective Naturalness test.

For DRT and subjective naturalness test results, 1-tailed paired t-tests are performed and the results are summarized in Table 6.3. A significance level of 0.05 is used for the statistical test. Given the low p-values for the DRT results (0.01) and Naturalness results (< 0.001), there is strong evidence that, on average, the use of the non-linear formant trajectory model does lead to significant improvements in both intelligibility and naturalness of the synthetic speech.

As for objective naturalness test as shown in Figure 6.5, a series of 1-tailed paired t-tests were conducted and these results are presented in Table 6.4. Many different types of models are considered in these results, such as male/female, CI/CD, PC/TJ (piecewise constant/non-linear trajectory models) and models with different number of states. As can be seen, when models are based on 3-state per phone, the use of a non-linear trajectory model leads to

Model set/State No.		3	4	5	6
Male	CI-PC vs. CI-TJ	$< \mathbf{0.01}$	$< \mathbf{0.01}$	$< \mathbf{0.01}$	$< \mathbf{0.01}$
	CD-PC vs. CD-TJ	$< \mathbf{0.01}$	0.18	0.09	$< \mathbf{0.01}$
	CD-PC vs. CI-TJ	$< \mathbf{0.01}$	0.18	0.56	$< \mathbf{0.01}$
Female	CI-PC vs. CI-TJ	$< \mathbf{0.01}$	0.08	0.09	0.32
	CD-PC vs. CD-TJ	$< \mathbf{0.01}$	$\mathbf{0.03}$	$\mathbf{0.02}$	0.36
	CD-PC vs. CI-TJ	$\mathbf{0.03}$	$< \mathbf{0.01}$	0.24	0.51

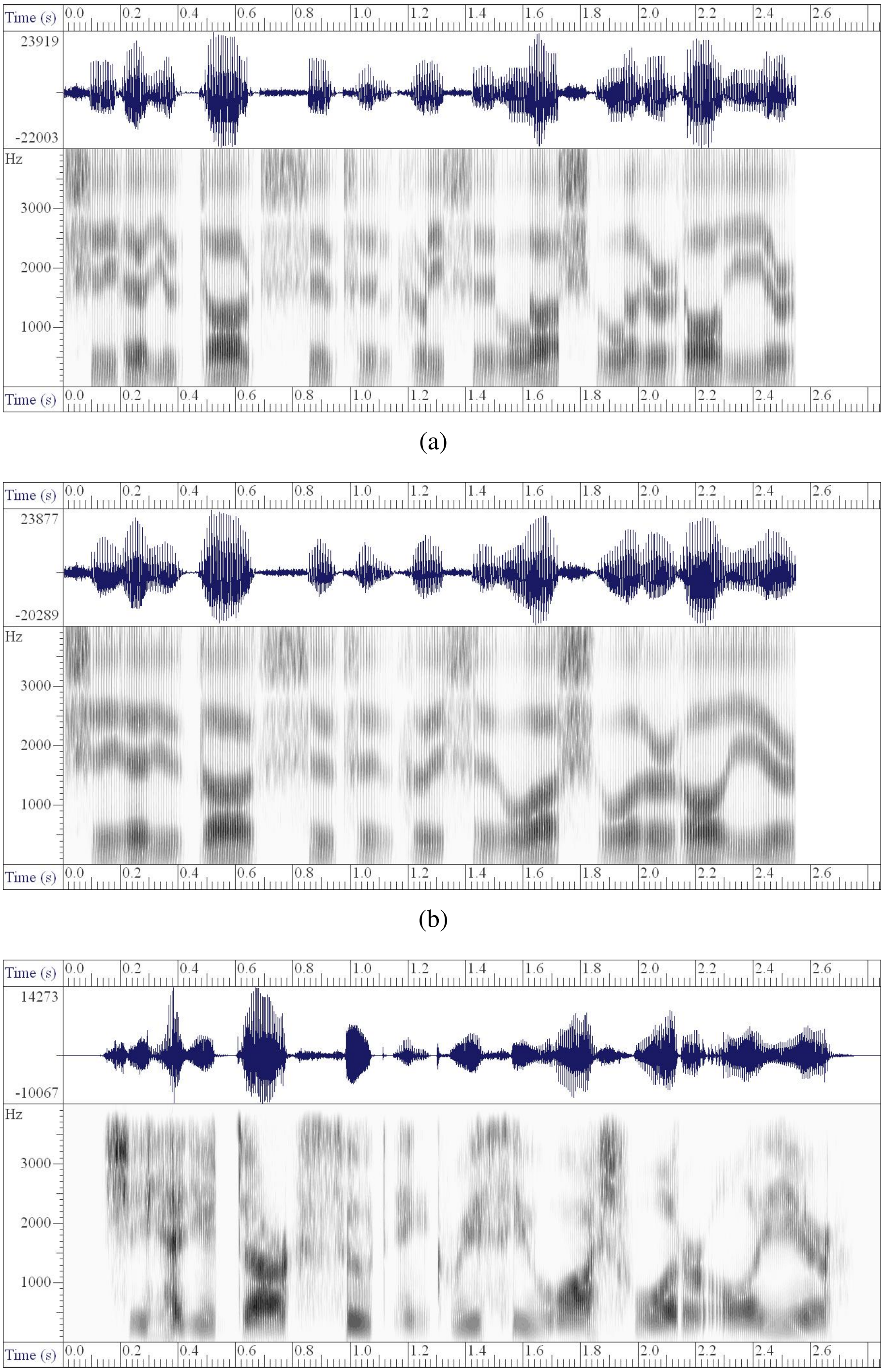
Table 6.4: t-test p-values for the objective naturalness test results as shown in Figure 6.5. CI/CD denote context-independent or -dependent models, and PC/TJ refers to piecewise contant or non-linear trajectory models. Numbers in boldface indicates that there is significant difference between the two results given a significance level of 0.05.

significant improvement in quality of the synthetic speech, relative to a piecewise constant model, for both CI and CD models, and both male and female speakers. More importantly, we have observed that, as can be seen in Figure 6.5, a non-linear trajectory CI model (denote as CI-TJ) outperforms the corresponding piecewise constant CD model (or CD-PC) in the naturalness test. In the significance test reported here, these improvements are proved to be statistical significant, as shown in Table 6.4 (see the rows corresponding to CD-PC vs. CI-TJ for both male and female). This is an encouraging result, as it shows that the CI-based non-linear trajectory models can not only model coarticulation, but also work better than the conventional CD models, with only a fraction of parameters of the CD models.

As has been discussed before, increasing the number of states per phone can improve the quality of the synthetic speech, though this improvement is often compromised by the size of the available training data, especially when CD models are used. This is also reflected in the t-test results as shown in Table 6.4, as improvements (when moving from piecewise constant to non-linear trajectory models) in quality of the synthetic speech generated from models with more than 3 states per phone are often not significant.

A closer look at the non-linear trajectory

The power of the non-linear trajectory model is largely due to the smoothing effect of the speech parameter generation algorithm (see Section 5.1.1 for the smoothing in the SPG algorithm; Section 3.2.3 also describes the DP smoothing used in Holmes formant analyzer), which reduces most discontinuities, particularly those between the boundaries of adjacent



(a)

(b)

(c)

Figure 6.6: (a) Wide-band speech spectrogram for the synthetic speech generated by using piecewise constant PFS control parameters. (b) Wide-band speech spectrogram for the synthetic speech generated by using non-linear trajectory PFS control parameters. (c) Wide-band speech spectrogram for an original TIMIT utterance, spoken by speaker `mdab0`, down-sampled to 8 KHz. In all cases, the sentence is ‘She had your dark suit in greasy wash water all year’ (sa1).

phones, which typically occur in the synthetic speech generated by piecewise constant synthesiser control parameters, as can be seen in Figure 6.6(a). Figure 6.6(b) demonstrates improved modelling of formant transitions due to the use of the SPG algorithm, as the formant trajectories become smoothed and continuous, relative to the corresponding formant trajectories shown in Figure 6.6(a), which are based on the piecewise constant model.

The speech parameter generation algorithm gives smooth trajectories, but it remains unclear whether or not they match actual formant trajectories in real speech. A more detailed comparison between actual formant trajectories in real speech and those generated by using the SPG algorithm is performed and depicted in Figure 6.7. The example real speech utterance is taken from TIMIT – "Are you looking for employment?" (sx229), spoken by speaker mdab0. The actual formant trajectories, indicated by the thin lines in Figure 6.7, are obtained by using the Holmes formant analyser, as described in Section 3.2.3. The thick lines in Figure 6.7 represent formant trajectories which are automatically generated from HMMs by using the SPG algorithm. Formant trajectories for the first three formant frequencies F_1, F_2 and F_3 are all shown in this figure.

As can be seen in Figure 6.7, the formant trajectories generated from HMM by using the SPG algorithm are continuous and, in general, smoother than actual formant trajectories, though in many cases they are fairly close, especially for the second formant frequency F_2. Conventional HMMs use CD models to account for contextual effects. It should be noted that the non-linear trajectories shown in Figure 6.7 are generated from CI models, and the example utterance is taken from the test set so that it was not used to train the HMMs. For these reasons, it is safe to say that the improved modelling of formant dynamics is due to the use of the non-linear formant trajectories generated by the SPG algorithm. So far, we have some experimental evidence that the use of non-linear formant trajectory is able to account for coarticulation effects from the perspective of speech synthesis, and this will be investigated further in the speech recognition experiments described in the next chapter.

For DRT, which tests the intelligibility of an individual consonant sound, the overall improvement resulted from the use of non-linear trajectory models is not as significant as

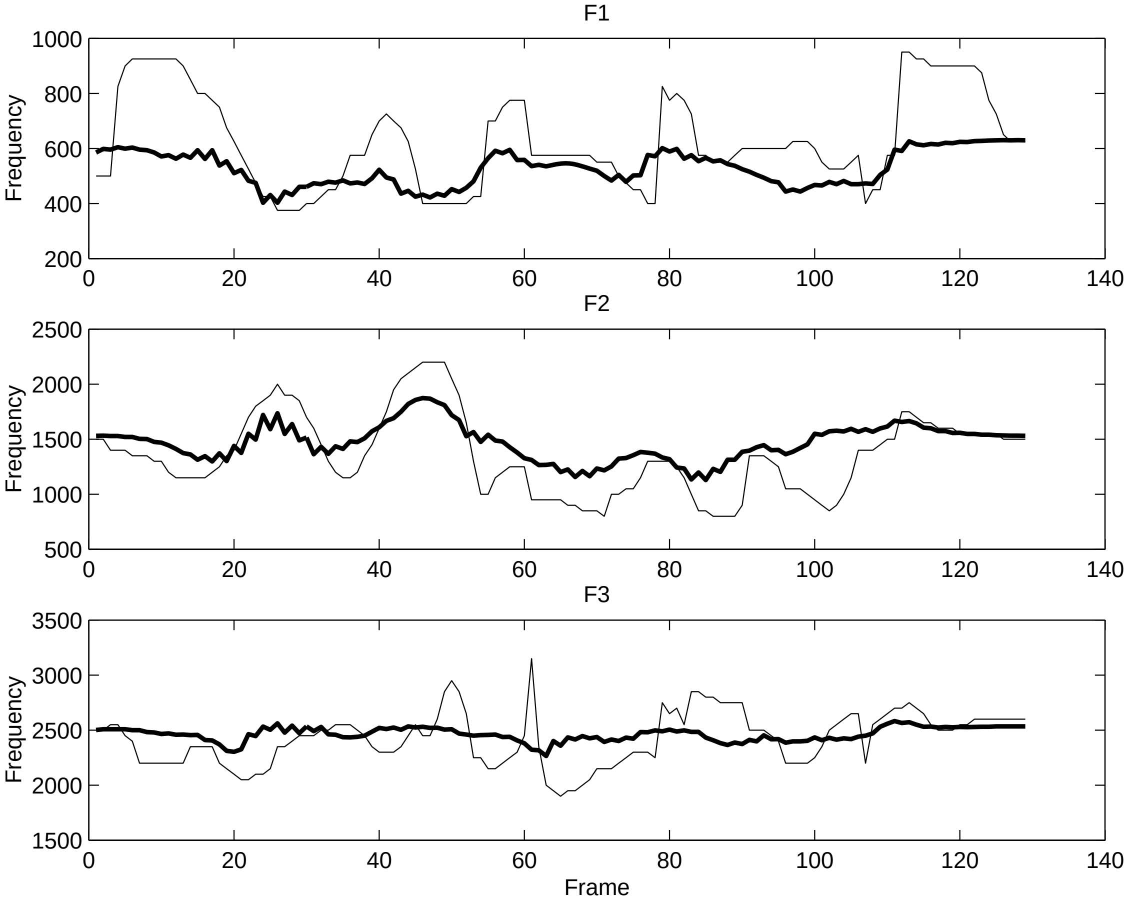


Figure 6.7: The first three formant trajectories in real speech and synthetic speech. In all sub-figures, the thin lines indicate formant trajectories in real speech (calculated by using the Holmes formant analyser), while the thick lines represent formant trajectories generated from HMMs by using the speech parameter generation algorithm. The real speech is a TIMIT utterance “Are you looking for employment” (`sx229`), spoken by speaker `mdab0`.

that of the naturalness test. In DRT, the synthesis is based on isolated words, which usually consists of just three phonemes. The smoothing effect of the speech parameter generation algorithm may be more easily overshadowed by other factors, for example noise or poor quality, than in a longer context (e.g., at the sentence level). Moreover, the speech generation algorithm can sometimes be too ‘good’ that over-smoothed trajectories are generated, which results in ‘muffled’ synthetic sounds (Black et al. 2007). This could have an adverse effect on the intelligibility test.

The effect of over-smoothing can be also seen in Figure 6.7, which exhibits a rather flat formant trajectory for F_3. Although the ‘over-smoothed’ sounds may not be a problem in the naturalness test, they may cause confusion in the intelligibility test as the realisation of each individual phone segment is affected excessively by the neighbouring phones. Therefore, the identity of the target phone may be blurred. In this case, the piecewise constant synthetic speech may be more intelligible because the synthetic sound is not affected by its context, though it may sound more robotic. To reduce the over-smoothing effect, a modified SPG algorithm based on the global variance was proposed (Toda & Tokuda 2001).

Correlation between subjective and objective measures

Both subjective and objective (PESQ) measures have been used to assess the naturalness of the synthetic speech, as shown in Figure 6.4. The objective measure chosen for the naturalness test is PESQ, which has been successfully used to evaluate speech quality for a wide range of conditions. PESQ results have shown high correlation with subjective MOS for various types of distortions (Rix et al. 2001, Hu & Loizou 2008). PESQ has also been used to evaluate the intelligibility of the synthetic speech in HMM synthesis (Valentini-Botinhao, Yamagishi & King 2011), though it is more successful in predicting quality.

Figure 6.8 presents two scatter plots to aid the analysis of the correlation between subjective and objective evaluation results on the quality of the synthetic speech. The top plot shows the naturalness test results based on the piecewise constant model, and the bottom plot is based on results for the non-linear trajectory model. Both plots show positive correlation between MOS and PESQ-LQO scores. In addition, Pearson's correlation coefficients are calculated based on the following formula

$$r = \frac{\sum_{n=1}^{N}(X_n - \bar{X})(Y_n - \bar{Y})}{\sqrt{\sum_{n=1}^{N}(X_n - \bar{X})^2 \sum_{n=1}^{N}(Y_n - \bar{Y})^2}}, \tag{6.5}$$

where X_n and Y_n are the subjective and objective scores respectively, $\bar{X}$ and $\bar{Y}$ the average subjective and objective scores respectively, and $N = 20$ is the sample size. The correlation coefficients for the piecewise constant model (corresponding to the top plot in Figure 6.8) and the non-linear trajectory model (the bottom plot in Figure 6.8) are 0.67 and 0.78, respectively. These results confirm the positive correlation between the subjective and objective results.

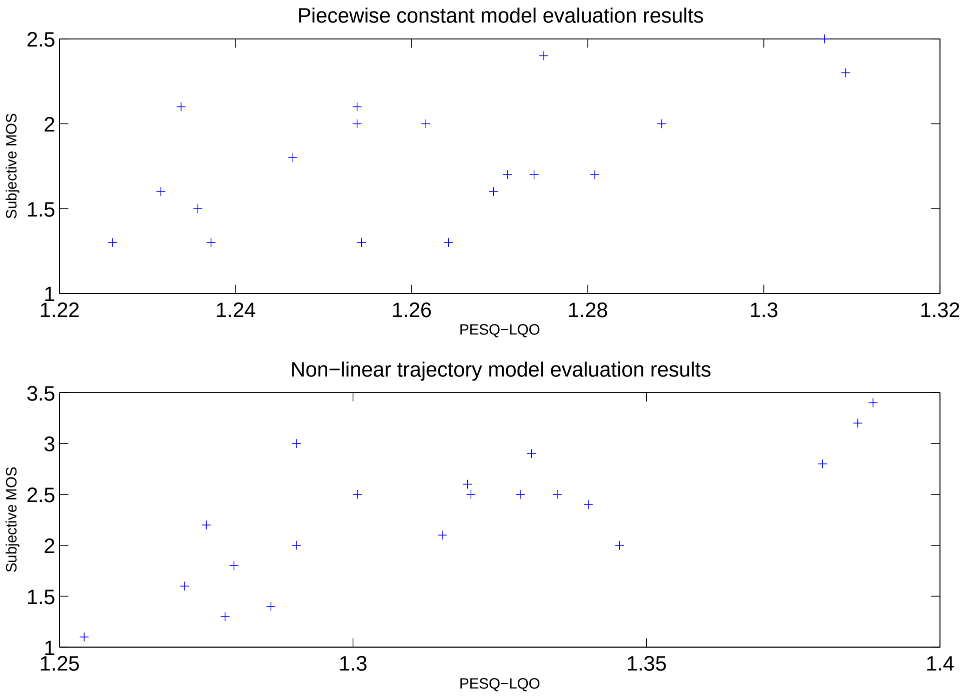
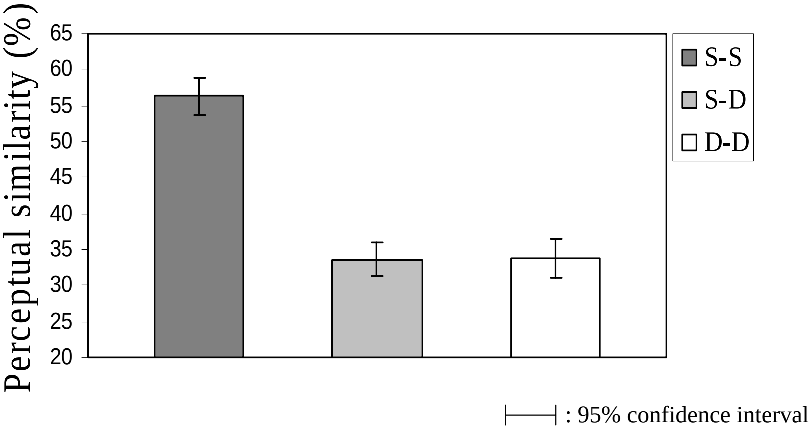


Figure 6.8: Scatter plots of subjective and objective results for the naturalness test for synthetic speech generated using the piecewise constant model (top plot) and the non-linear trajectory model (bottom plot).

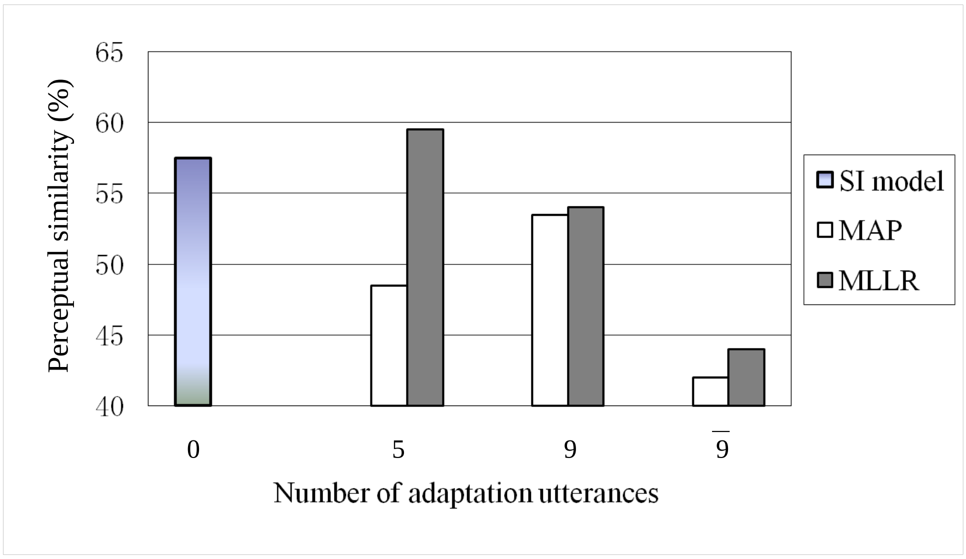
6.3.4 Mimicry test results

The mimicry test results are summarized in Figure 6.9. Figure 6.9(a) shows the perceptual similarity between an original utterance and a semi-synthetic utterance. The meaning of the semi-synthetic speech is explained in Section 6.2.5 on page 114. It can be seen from Figure 6.9(a) that, even for the first group (S-S), i.e., the same speaker saying the same utterance, the similarity score between original speech and semi-synthetic speech is 56.8%, which suggests that the semi-synthetic speech has already lost some information about a speaker's voice characteristics. In addition, it can be seen that there are no significant differences between the test results for groups S-D and D-D, with scores of 33.5% and 33.6% respectively. This indicates that the perceptual similarity between an original utterance and a semi-synthetic utterance drops sharply when the utterances are different (i.e., the content of the sentence), irrespective of whether they are from the same or different speakers.

Figure 6.9(b) shows the perceptual similarity between semi-synthetic speech and syn-



(a)



(b)

Figure 6.9: (a) Perceptual similarity between original speech and semi-synthetic speech. S-S: same speaker - same utterance; S-D: same speaker - different utterances; D-D: different speaker - different utterances. (b) Perceptual similarity between semi-synthetic speech and synthetic speech based on HMMs adapted to the same and different speakers. $\bar{9}$: HMMs adapted to a different speaker with 9 adaptation utterances.

thetic speech based on HMMs adapted to the same and different speakers with different numbers of adaptation utterances. Number '0' means there is no adaptation; that is, speaker independent models are used to generate synthetic speech. First of all, Figure 6.9(b) shows that the perceptual similarity falls when the model set is adapted to a different speaker (denoted by $\bar{9}$), which was expected before the test. However, the results of the mimicry test using MLLR and MAP adapted to the same speaker are not expected. In the case that MLLR is used for speaker adaptation, the perceptual similarity score first increases when moving from SI models to SD models with 5 adaptation utterance, then decreases as the number of adaptation utterances increases from 5 to 9. When the MAP adaptation technique is used, the similarity falls from 57.5% to 48.5% as the number of adaptation utterance turns from 0 (i.e., SI models) to 5, and then increases to 53.5% when 9 adaptation utterances are used.

6.3.5 Mimicry test results analysis

As has been explained in Section 6.2.5, the goal of the similarity test is to determine whether the two utterances are spoken by the same speaker or not. The participants were therefore told to focus on the voice characteristics of the speakers, rather than the sentences that they heard. However, it was found that direct comparison between the original speech and the synthetic speech was very difficult due to the poor quality of the synthetic speech. Hence semi-synthetic speech was introduced as an intermediate type of speech for comparison between the original speech and the synthetic speech.

Original v.s. Semi-synthetic analysis

The similarity test result between the original speech and the semi-synthetic speech becomes interesting due to the use of different utterances for original and semi-synthetic speech. First of all, the first and second column in Figure 6.9(a) demonstrate how the use of different utterances affects the similarity test result. The similarity score decreases significantly when the utterances are different for the original speech and semi-synthetic speech, even though they are indeed from the same speaker. This seems to suggest that the content (or what is being said) has a significant effect in determining the speaker identity.

Furthermore, the second and third column in Figure 6.9(a) show the similarity results of either the same or different speakers saying different utterances for original speech and semi-synthetic speech respectively. It can be seen that these two similarity scores are virtually the same. That is to say, the change of the speaker identity (hence the change of speaker voices) has no effect for this type of similarity test.

The above dilemma may be accounted for in the following two aspects. Firstly, it might be that the quality of the semi-synthetic speech is not good enough to preserve the important information about the speaker's voice characteristics. The result from the first column in Figure 6.9(a) proves this to some extent, with a similarity score of only 57%. However, the semi-synthetic speech is generated from a sequence of synthesiser control parameters which are derived from the original speech using the Holmes formant analyser. In principle, the synthesiser control parameters obtained in this way (also known as copy synthesis) should be fairly good. Secondly, given the poor quality of the synthetic speech, the participants may be overwhelmed by the effect of the use of different utterances and could not make sensible judgments on the speakers' voice characteristics. Therefore, even though the speaker is different, as shown in the second and third column in Figure 6.9(a), the similarity results remains approximately the same, due to the use of different utterances.

Recently, NIST has conducted a test of Human Assisted Speaker Recognition (HASR) (Greenberg, Martin, Doddington & Godfrey 2011), which involves a limited number of challenging trials (15 in HASR1, 150 in total), which are selected carefully by automatic processing and human listening. Each trial consists of two speech audio recordings, including an interview recorded over a microphone channel, and a phone conversation recorded over a telephone channel, both collected by the LDC. These recordings contain different channels, different speaking styles and obviously, different content. An participant is required to make a decision (true or false) on whether the same speaker was presented in both of these two recordings in order to complete a trial. In addition, the participant needs to give a confidence score on the choice, where a higher score indicates that the participant has greater confidence that the same speaker was presented in both audio recordings. The trial-by-trial results of the 15 HASR1 trials for all 20 participating systems are summarised in (Greenberg et al. 2011). These results clearly show that, in difficult cases, humans have trouble making the correct choice. For example, a majority of the 20 HASR systems had an incorrect decision for 5 of the 15 trial. In addition, it is also noted that automatic speaker recognition systems outperform the HASR systems on the same HASR2 trials.

The NIST HASR results suggest that humans are not as good as one might expect in recognising who is speaking, especially under difficult conditions. The mimicry test described in this chapter is similar to the NIST HASR test in that both tests focus on predicting whether the two audio recordings are spoken by the same speaker or not. In either test, the test sentences spoken by the speaker(s) are different. The major difference between the NIST experiment and the mimicry test is that only real human spoken utterances are used in the NIST test, while synthetic utterances are also used in this mimicry test. The NIST results show that humans often have difficulty in recognising speakers, even though two real human speech utterances are provided. It is therefore not surprising that it becomes more difficult to recognise speakers given a synthetic speech, as shown in this mimicry test.

Semi-synthetic v.s. Synthetic analysis

The similarity test results between semi-synthetic and synthetic speech are presented in Figure 6.9(b) and analysed as follows.

MLLR v.s. MAP For speech recognition, MLLR generally performs better with small amount of adaptation data and MAP adaptation gradually catches up when more adaptation data is available (Young et al. 2005). This 'rule of thumb' seems still applicable in this similarity test between semi-synthetic speech and synthetic speech, in which MLLR-based synthetic speech outperforms MAP-based synthetic speech, with at most a 10% difference between the similarity scores (when 5 adaptation utterances are used), as shown in Figure 6.9(b).

The use of different speakers data Initially, the motivation of using 9 adaptation utterances from a different speaker is to contrast the result of using the adaptation data from the same speaker. It was expected that the use of 9 adaptation utterances from a different speaker would result in a drop in the similarity score. The similarity test results shown in Figure 6.9(b) confirm this, illustrated as the case $\bar{9}$, where the model set is adapted to another speaker using 9 adaptation utterances from that speaker. Actually, for either MLLR or MAP adaptation, the similarity score of using 9 different adaptation utterances is the lowest, as shown in Figure 6.9(b).

Unexpected drop in the similarity scores It can be seen from Figure 6.9(b) that the increase of the number of adaptation utterances from the same speaker results in decreasing similarity scores, while it was expected that the similarity would increase when more adaptation data is available. In case the MAP adaptation is used, the similarity first decreases when 5 adaptation utterances are used and then increases as the number of adaptation utterances increases from 5 to 9.

The drop in the similarity scores with increasing number of adaptation utterances is unexpected. A possible explanation is that the amount of adaptation data (less than 10 utterances) is too small so that the overall trend of the change in similarity with respect to the number of adaptation utterances has not yet emerged with less than 10 adaptation utterances. The increase in the similarity score as the number of adaptation utterances increases from 5 to 9 for the case of MAP adaptation is a first sign. However, the limitation on the maximum number of adaptation utterances is due to the TIMIT speech corpus used in the experiment, in which each speaker speaks at most 10 utterances.

A similar subjective similarity experiment was conducted and reported by (Yamagishi & Kobayashi 2007*b*). Five-point scale was used, i.e., Very similar, Similar, Slightly Similar, Dissimilar, Very Dissimilar. MLLR was employed in their experiment to adapt the so-called average-voice-based model (Yamagishi 2006) to a target speaker. The number of adaptation utterances varied from 5, via 10, 20, 50, 100, 150, 250, 350 to 450. The authors found that approximately 100 utterances were required to represent the target speaker and synthesise speech appropriately in their HMM-based speech synthesis system. When the number of adaptation utterances was smaller than 20, the average score was below the 'slightly similar' scale, which comply with the results reported here.

It is interesting to compare these results with that of speech recognition. For speech recognition, only 3 adaptation utterances (equivalent to an average of 11 sec of speech per speaker) can result in improvement in recognition rates using a global transformation in MLLR adaptation (Leggetter & Woodland 1995). For HMM-based speech synthesis, it might be the case that more adaptation utterances should be available before the trend of the change in the perceived similarity becomes stable.

Despite the above argument on the lack of adaptation data, it should be confessed that the quality of the synthetic speech plays an important role in all subjective listening experiments. For example, the similarity test results might be different, provided that the quality of the synthetic speech is good. However, the effort that would be involved in building a better synthesiser is beyond the project.

Evaluating model adaptation using PESQ

An objective measure may be useful to aid the analysis of model adaptation for HMM-based speech synthesis. As in the case of the naturalness test, PESQ is employed to evaluate synthetic speech generated from SI and SD models with different number of adaptation utterances. The test data is the same as that used in the 'semi-synthetic vs. synthetic' experiment. Figure 6.10 shows the PESQ results on synthetic speech generated from adapted models with respect to different number of adaptation utterances.

It should be noted that this is not an 'objective version' of the mimicry test but an objective quality test. The focus of this analysis is on the relationship between model adaptation and the quality of the synthetic speech. As is well known, model adaptation has been successful in improving speech recognition performance. Figure 6.10 clearly shows that model adaptation is also useful in improving the quality of the synthetic speech. When models are adapted using data from the same speaker, increasing the number of adaptation utterances

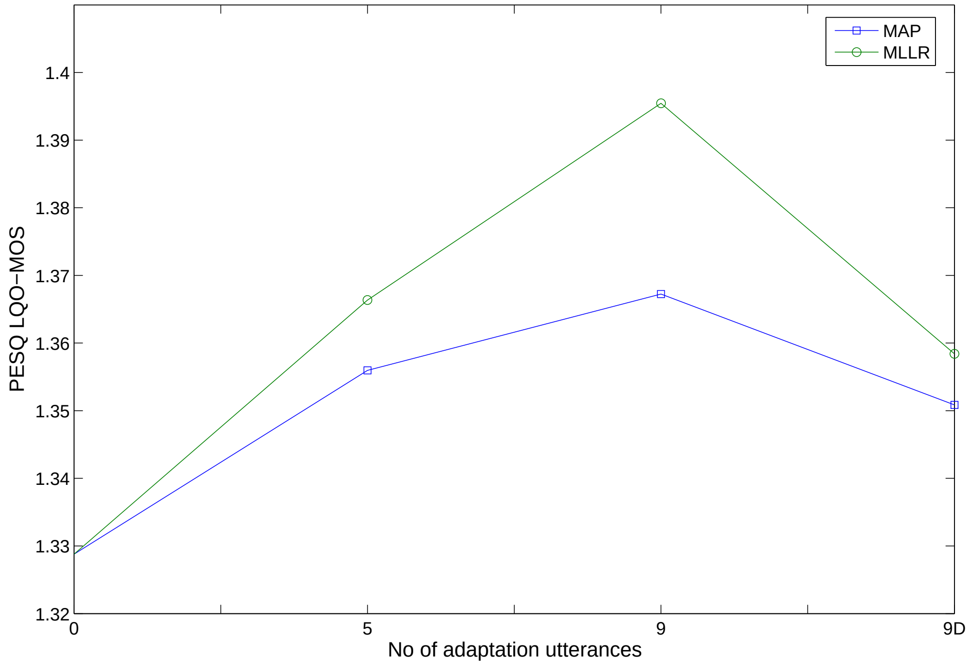


Figure 6.10: Objective evaluation results on model adaptation based on PESQ.

(from 0 to 9) leads to consistently improved quality of the synthetic speech. In addition, MLLR based adaptation works consistently better than MAP adaptation. Finally, it is interesting to see that when the models are adapted to another speaker (corresponding to '9D' at the right side of Figure 6.10), PESQ score also drops, as in the case of the subjective mimicry test.

6.4 Summary

This chapter describes the research on modelling formant dynamics using a non-linear trajectory model generated by the SPG algorithm. This non-linear trajectory model of formant dynamics is assessed using the HMM-base speech synthesis paradigm. A basic HMM-based TTS system was built on 12 PFS control parameters and their time derivatives. Two kinds of synthesiser control parameter sequences were generated from HMMs, namely, the piecewise constant PFS control parameters and non-linear trajectory PFS control parameters. These two types of synthesiser control parameter sequences were used to synthesise speech from the Holmes parallel formant synthesiser.

Three types of human listening tests were conducted to assess the performance of the non-linear trajectory model. In addition, PESQ is employed as an objective measure to evaluate the quality of the synthetic speech. Twenty subjectives participated in the subjective experiments. Despite the poor overall quality of the synthetic speech, the DRT and naturalness test results show that the use of the non-linear trajectory model of formant dynamics leads to improvement in both intelligibility and overall quality of the synthetic speech, relative to the piecewise constant model. Statistical significance tests confirmed that these improvements are significant. The non-linear formant trajectories generate from HMMs have been shown to match the actual formant trajectories in real speech fairly well, though sometimes over-smoothing may occur. These results suggest that the non-linear trajectories generated by using the parameter generation algorithm is able to account for contextual effects, such as coarticulation.

Conventional speaker adaptation techniques, including MLLR and MAP, were employed to generate adaptable synthetic speech. The similarity experiment attempted to test the similarity of the voice characteristics between a real utterance and an adapted, synthetic speech. However, experiment results show that these techniques (MLLR and MAP) are not particularly successful under this setting. Two main reasons have been spotted. Firstly, the low quality of the synthetic speech poses a very challenging task for human listeners. Research elsewhere indeed found that humans have trouble recognising speakers in difficult situation. The second possible reason is the limitation on the amount of available adaptation data, which is imposed by the particular corpus used in the test.

Chapter 7

Recognition using non-linear trajectories

7.1 Introduction

The previous chapter shows that the use of a non-linear trajectory model of formant dynamics can improve the performance for speech synthesis, compared to a piecewise constant trajectory model. This chapter continues to assess the non-linear trajectory model of dynamics from the perspective of speech recognition. In a linear/linear MSHMM presented in (Russell & Jackson 2005, Russell et al. 2007), formant dynamics are modelled as piecewise linear trajectories and transformed into the acoustic layer using a set of linear articulatory-to-acoustic mappings. For a given representation of the intermediate layer, it was hoped that further improvements in recognition performance would be achieved by using non-linear formant trajectories in the intermediate layer of MSHMMs.

As in the speech synthesis experiments described in Chapter 6, non-linear formant trajectories are generated based on the speech parameter generation algorithm (Tokuda, Kobayashi & Imai 1995, as described in Section 5.2). The formant-to-acoustic mappings remain linear in this research, which is estimated based on matched sequences of acoustic and formant data, as described in (Russell & Jackson 2005, see Section 4.4.1 on page 75). However, the mapping parameters are re-estimated using the non-linear formant trajectories in order to provide better fit between acoustic data and formant data. The resulting model is referred to as a non-linear/linear MSHMM, and is represented in Figure 7.1.

As in (Russell & Jackson 2005, Russell et al. 2007), five ‘formant-to-acoustic’ map-

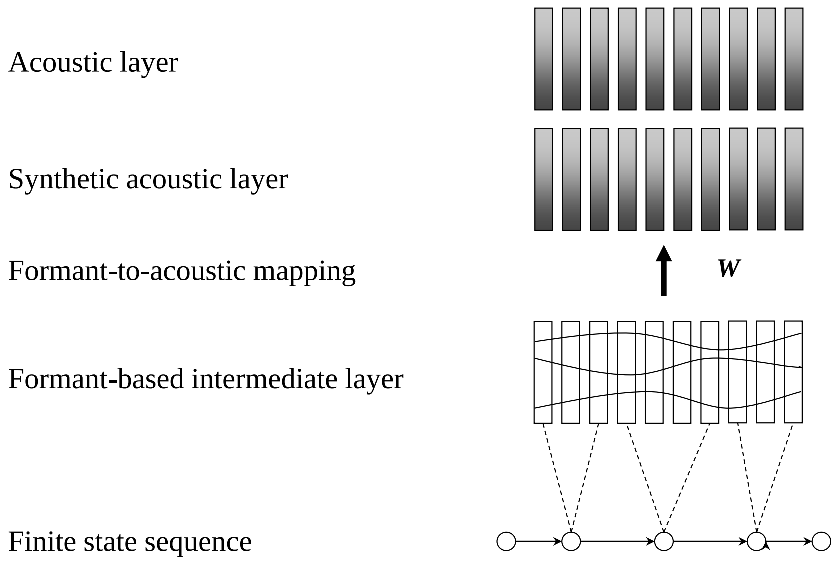


Figure 7.1: A non-linear/linear MSHMM in which speech dynamics are modelled as non-linear formant trajectories in the intermediate layer, and transformed into acoustic layer using linear formant-to-acoustic mapping.

ping schemes (see Table 4.1 on page 81), based on different phone partition schemes, are considered in order to investigate their effects on phonetic recognition performance. The intermediate layer of the MSHMM, though, is restricted to the 12PFS representation only in experiments described in this chapter.

The rest of this chapter is structured as follows. Section 7.2 describes the N-best list rescoring scheme that is used to evaluated the new model. The MSHMM model sets used in the rescoring experiments and model parameter esitmation are described in Section 7.3. Section 7.4 provides a detailed description of the rescoring experiments. The rescoring results and analysis are reported in Section 7.5.

7.2 N-best list rescoring

N-best list rescoring is commonly used when developing speech recognition systems. The existence of an N-best list enables us to combine additional knowledge sources, such as complicated acoustic and language models, into the recognition process (Ostendorf 1991). An N-best list contains N candidate recognition hypotheses for each test utterance, which can be rescored and reordered using additional knowledge sources. N-best list rescoring can be used to evaluate a new technique before investing the effort to implement a decoder. Alternatively, a decoder may not be practically feasible. This research resorts to N-best list rescoring to evaluate the new non-linear/linear MSHMMs.

7.2.1 N-best list rescoring overview

In the context of statistical modelling of speech patterns, the recognition problem is, in general, to find the most likely word string $\hat{\mathcal{W}} = \{w_1, \ldots, w_K\}$ given a sequence of acoustic observations $\boldsymbol{O} = \{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_T\}$, such that the probability $P(\mathcal{W}|\boldsymbol{O})$ of the word sequence $\mathcal{W}$ given the acoustic evidence $\boldsymbol{O}$ is maximised

$$\hat{\mathcal{W}} = \arg\max_{\mathcal{W}} P(\mathcal{W}|\boldsymbol{O}) = \arg\max_{\mathcal{W}} \{p(\boldsymbol{O}|\mathcal{W})P(\mathcal{W})\}, \tag{7.1}$$

where $p(\boldsymbol{O}|\mathcal{W})$ and $P(\mathcal{W})$ are determined by an acoustic model and a language model, respectively. In an N-best list $\widetilde{\mathcal{W}} = \{\tilde{\mathcal{W}}_1, \ldots, \tilde{\mathcal{W}}_N\}$, each entry $\tilde{\mathcal{W}}_n$ $(1 \leq n \leq N)$ represents one possible interpretation of the unknown observation sequence $\boldsymbol{O}$.

Figure 7.2 shows the N-best list rescoring paradigm adopted in this research and the relationships between different model sets and the N-best list. On the left side of Figure 7.2 is a set of conventional, triphone HMMs. The main role of these conventional HMMs in the rescoring experiments is to generate an N-best list for each test utterance. In addition, these standard HMMs can be used to produce an optimal state alignment for each hypothesis in the N-best list. The MSHMMs on the right side of Figure 7.2, including both linear/linear MSHMMs and non-linear/linear MSHMMs, are used to rescore the N-best list.

The N-best rescoring is conducted as follows. Given an N-best list $\widetilde{\mathcal{W}} = \{\tilde{\mathcal{W}}_1, \ldots, \tilde{\mathcal{W}}_N\}$ for a test utterance $\boldsymbol{O}$, a hypothesis $\tilde{\mathcal{W}}_n$ $(1 \leq n \leq N)$ from the N-best list is taken and the probability $p(\boldsymbol{O}|\tilde{\mathcal{W}}_n)$ of the test utterance given the hypothesis is calculated (the probability calculation is described in Section 7.2.3). The probability calculation is repeated for every

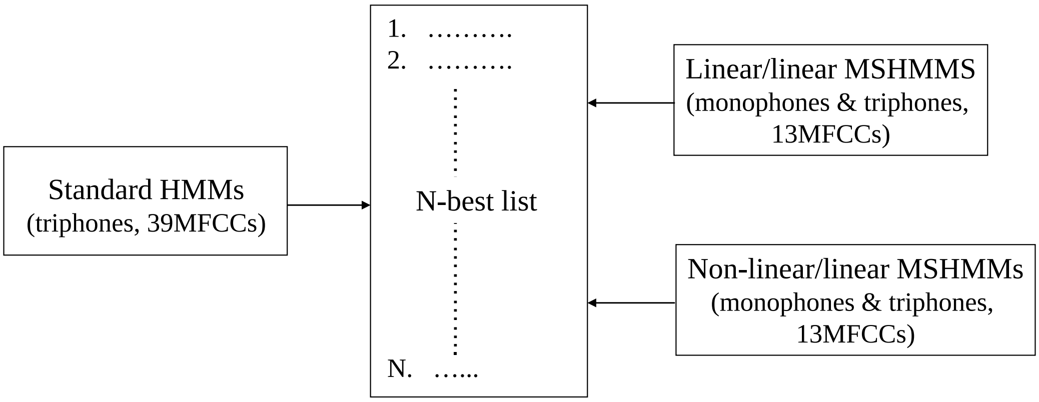


Figure 7.2: A schematic diagram of the N-best list rescoring paradigm.

hypothesis in the list $\widetilde{\mathcal{W}}$ and the results are recorded. A search is then made to find the hypothesis which gives rise to the highest probability. Then the N-best list is reordered so that this hypothesis is placed at the top of the list. This process is repeated for every test utterance in the test set and its corresponding N-best recognition hypotheses. Once every N-best list for every test utterance in the test set has been processed, the 'top' hypotheses in the N-best lists for all test utterance are taken for further analysis.

7.2.2 N-best list generation

Two N-best lists are generated and used for rescoring in this research, and both lists are generated using triphone HMMs with HTK.

1000-best list

The acoustic models used to generate the 1000-best list are gender-dependent[1] triphone HMMs, which are trained on data from all male speakers in the TIMIT training set, including 3260 utterances. The test set comprises all recordings of male speakers in the TIMIT core test set (including 16 subjects, 128 utterances, excluding 'sa1' and 'sa2' sentences). Acoustic features used in generating the 1000-best list are 13MFCCs (including the zeroth) plus Δ

[1] See Section 3.3 on page 66 for the motivation for gender-dependent models.

and Δ^2 coefficients (25ms window, 10ms fixed frame rate). The calculation of MFCCs is described in Section 3.1 and performed using HTK.

100-best list

The 100-best list is generated from triphone HMMs trained on data in the TIMIT full training set, including 4620 training utterances for both male and female speakers. The TIMIT core test set is used, including 16 males and 8 females, 192 utterances. As in the generation of the 1000-best list, 13MFCCs plus Δ and Δ^2 coefficients are used as acoustic features.

TIMIT bigram language model

A bigram language model is used in the generation of the N-best list. In a general bigram language model, the probability of the kth word w_k in the word sequence $\mathcal{W}$ depends only on the identity of the immediate previous word w_{k-1}

$$P(w_k|w_1, w_2, \ldots, w_{k-1}) \approx P(w_k|w_{k-1}). \tag{7.2}$$

The language model probability for the word sequence $\mathcal{W}$ is calculated as follows:

$$P(\mathcal{W}) = P(w_1)\prod_{k=2}^{K} P(w_k|w_1, \ldots, w_{k-1}) = P(w_1)\prod_{k=2}^{K} P(w_k|w_{k-1}). \tag{7.3}$$

In principle, n-grams can be estimated using simple frequency counts of occurrences in the training database. For example, the bigram probability $P(w_k|w_{k-1})$ for each word pair w_{k-1} and w_k is given by

$$P(w_k|w_{k-1}) = \frac{N(w_k, w_{k-1})}{N(w_{k-1})} \tag{7.4}$$

where $N(w_k, w_{k-1})$ is the number of times that the word pair (w_{k-1}, w_k) occurs in all the training label files, and $N(w_{k-1})$ denotes the number of occurrence of the word w_{k-1} in the training text.

Data sparsity is an ever-present problem in language modelling when using n-grams, because even for small vocabularies, many n-grams will either do not appear in the training material or do not occur frequently enough to enable robust estimation of their probabilities. Backing off is commonly used to deal with the data sparsity problem, in which the probability for an unseen n-gram is replaced with a scaled lower order model probability. For example, using back off, the probability of a missing trigram is replaced with a rescaled bigram probability

$$\hat{P}(w_k|w_{k-1}, w_{k_2}) = B(w_{k-1}, w_{k_2})P(w_k|w_{k-1}), \tag{7.5}$$

where $B(w_{k-1}, w_{k_2})$ is a back off function included to ensure that $\hat{P}(w_k|w_{k-1}, w_{k_2})$ is properly normalised (Young 1995).

This research uses a *phone* level bigram language model with backing off, which is estimated using all of the TIMIT phonetic label files in the training set using HTK.

N-best list generation and evaluation

N-best list type	N-best decoding	Phone accuracy (%)	Sentence correct (%)
1000-best list	1-best	65.7	0
	1000-best	77.9	0.78
100-best list	1-best	70.6	0
	100-best	77.3	0.52

Table 7.1: N-best list evaluation results for the 1000-best list and 100-best list.

Given a set of triphone HMMs and a TIMIT bigram language model, the N-best list is generated using the HTK decoder for each test utterance. For comparison purpose, some preliminary experiments are conducted and results are shown in Table 7.1. The baseline phonetic recognition phone accuracies (i.e., the 1-best decoding) are 65.7% and 70.6% for the 1000-best list and the 100-best list, respectively. The upper bound of phone accuracy for the 1000-best list is 77.9%, which is based on the most accurate match among the 1,000 alternatives and the corresponding reference label file. Therefore, this is the theoretical maximum score the following N-best rescoring experiments can reach. Similarly, for the 100-best list, this theoretical upper bound is 77.3%. Moreover, the sentence correct for the 1000-best list, based on the most accurate match, is 0.78%. In fact, only 1 correct label file at sentence-level appears in the 1000-best lists. In the case of the 100-best list, there is also just 1 correctly recognised sentence, which gives a sentence correct of 0.52%.

7.2.3 Segment probability calculation in MSHMMS

This section describes the probability calculation for each test utterance given a hypothesis in the N-best list using either a linear/linear MSHMM or a nonlinear/linear MSHMM. Suppose that an unknown test utterance corresponds to a sequence of acoustic feature vectors $\boldsymbol{O} = \{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_T\}$. For a given hypothesis $\tilde{\mathcal{W}}_m$ in the N-best list, this utterance can be explained via a state sequence $s = \{\tau_1 \times s_1, \tau_2 \times s_2, \ldots, \tau_N \times s_N\}$, where $\tau_1 + \tau_2 + \ldots + \tau_N = T$ and $\tau_n \times s_n$ denotes τ_n repetitions of state s_n. In the intermediate space of an MSHMM, a trajectory $\boldsymbol{f}$ of length T is defined as $\boldsymbol{f} = \{\boldsymbol{f}_1, \boldsymbol{f}_2, \ldots, \boldsymbol{f}_N\}$, where $\boldsymbol{f}_n$ is a trajectory 'fragment' of length τ_n, associated with state s_n. There are two types of trajectory, corresponding to linear/linear MSHMMs and non-linear/linear MSHMMs, respectively.

- In the case of a linear/linear MSHMM, each trajectory fragment $\boldsymbol{f}_n$ is a linear trajectory, defined by a slope vector and a mid-point vector for state s_n, as described in Section 4.2.

- For a non-linear/linear MSHMM, $\boldsymbol{f}$ is a non-linear, continuous trajectory obtained using the SPG algorithm based on the given state sequence s (See Section 5.2), and $\boldsymbol{f}_n$ is simply the section of the trajectory $\boldsymbol{f}$ which corresponds to state s_n.

In either case, the probability of the test utterance $\boldsymbol{O}$ given the hypothesis $\tilde{\mathcal{W}}_m$ can be calculated as

$$p(\boldsymbol{O}|\tilde{\mathcal{W}}_m) = \prod_{n=1}^{N} d_n(\tau_n) \Bigg(\prod_{t=1}^{\tau_n} \mathcal{N}\Big(\boldsymbol{o}_{\phi_n+t} | \boldsymbol{W}_{s_n}\big(\boldsymbol{f}_n(\phi_n + t)\big), \boldsymbol{\Sigma}_n\Big)\Bigg), \tag{7.6}$$

where $\phi_n = \sum_{i=1}^{n-1} \tau_i$ denotes the time spent before entering state s_n ($\phi_1 = 0$), d_n the duration PDF associated with state s_n, and $\boldsymbol{W}_{s_n}$ the formant-to-acoustic mapping associated with state s_n. Equation 7.6 forms the basis of the N-best list rescoring. In the rescoring experiments described in Section 7.4, this equation is used to compute the probability of each test utterance given each possible hypothesis in its corresponding N-best list for both linear/linear and non-linear/linear MSHMMs.

7.3 MSHMMs used in rescoring

This section describes various MSHMM models used in the N-best list rescoring experiments, including the training method of these models.

7.3.1 Speech data

Acoustic data

MSHMMs model parameters are trained on the static 13MFCCs (including zeroth), without dynamic features. For the 1000-best list rescoring system, the recordings of all male speakers in the TIMIT training set are used to train MSHMMs, comprising 3260 training utterances. For the 100-best list rescoring system, the full TIMIT training set is used to train MSHMMs, including 4620 training utterances. Up until now, only static acoustic features are used in the framework of MSHMMs, while conventional state-of-the-art HMM systems also employ dynamic features. This is because that an MSHMM is motivated by the belief that the explicit modelling of dynamics in the articulatory-related intermediate space would provide an improved modelling of speech dynamics and therefore the dynamic features would be not necessary. In addition, this is also one of the examples of speech modelling, rather than speech description(see Section 2.2.3 on the use of dynamic features in HMMs).

Formant data

The intermediate layer of the MSHMMs concerned in this work is restricted to the formant-based representation of the 12 PFS control parameters (See Table 3.1 on page 56 for the 12 PFS data), which are computed using the Holmes formant analyser and converted to HTK format (see Section 3.2.3 on page 59 for more details of the Holmes formant analyser). The Holmes formant analyser generates one 12-dimensional PFS vector every 10ms.

7.3.2 MSHMMs model sets

	linear/linear MSHMMs		non-linear/linear MSHMMs	
	monophone	triphone	monophone	triphone
1A	√	√	√	√
6B	√	√	√	√
10C	√	√	√	√
10D	√	√	√	√
49E	√	√	√	√

Table 7.2: A summary of the model sets used in the 1000-best rescoring experiments.

A variety of gender-dependent MSHMM model sets for male speakers are trained in the 1000-best list rescoring experiments, as summarised in Table 7.2. All five formant-to-acoustic mapping schemes described in (Russell & Jackson 2005) are considered in this research. (These mappings are summarised in Table 4.1 on page 81.) Under each formant-to-acoustic mapping scheme, a set of linear/linear MSHMMs are firstly built, including both CI and CD triphone models. Based on these linear/linear MSHMMs, corresponding non-linear/linear MSHMMs are derived by replacing the piecewise linear trajectories in the intermediate layer with non-linear formant trajectories.

For the 100-best list rescoring experiments, a similar set of models (as shown in Table 7.2) are built. However, this time models are trained based on the full TIMIT training set. In this case, the resulting models are gender-independent MSHMMs. This is the first time in the context of our MSHMM framework that gender-independent models are used. The motivation for the use of gender-dependent MSHMM has been stated earlier (see Section 3.3 on page 66), mainly to reduce the variability in speech signals. Here the motivation for using gender-independent MSHMMs is that the results may be put in a broader context, as many phone recognition work on TIMIT present results on the TIMIT core test set using gender-independent models. However, the use of data from both male and females speakers to train MSHMMs does introduce additional variability, which could potentially ‘hurt’ the performance.

7.3.3 Linear/linear MSHMMs

Unless otherwise stated, the MSHMM training described in this section applies to both gender-dependent or -independent MSHMMs.

Monophone MSHMMs training

The first step in building a CI monophone MSHMM system is to create a counterpart of a conventional HMM model set. A three-state, left-to-right (no-skip) conventional monophone HMM is built for each symbol in the TIMIT 49-phone set (see appendix A) using HTK. Each state in the HMM is associated with a single Gaussian PDF with a diagonal covariance matrix. The acoustic feature is 13MFCCs, without dynamic features. The model parameters of the conventional HMMs are optimised based on the Baum-Welch algorithm implemented in standard HTK tools, as described in Section 6.2.3.

Linear formant-to-acoustic mappings $\boldsymbol{W}_k$, where $k \in \{1, \ldots, K\}$, are then estimated using matched sequences of formant data (12PFSs) and acoustic data (13MFCCs) for each of the five mapping schemes, as described in Section 4.4.1. Given a conventional HMM M_ϕ, which represents a phone ϕ in phone class k, the corresponding MSHMM $\mathcal{M}_\phi$ is created and initialised in the following steps:

1. The mid-point vector $\boldsymbol{c}_j$ for state j of $\mathcal{M}_\phi$ in the formant-based articulatory domain is defined as $\boldsymbol{c}_j = \boldsymbol{W}_k^\dagger \boldsymbol{\mu}_j$, where $\boldsymbol{\mu}_j$ is the mean vector for state j of the HMM M_ϕ and $\boldsymbol{W}_k^\dagger$ is the pseudo-inverse of the formant-to-acoustic mapping $\boldsymbol{W}_k$ for phone class k.

2. The slope vector $\boldsymbol{m}_j$ for the jth state of $\mathcal{M}_\phi$ is set to zero.

3. The state duration PDF is uniform and the maximum state duration is set to 15 frames, i.e., $\tau_{max} = 15$, which is based on an analysis of the TIMIT transcription files and is sufficient to accommodate all phone labels in the TIMIT corpus.

4. The variance vector $\boldsymbol{\nu}_j$ for the jth state of $\mathcal{M}_\phi$ in the acoustic domain is set equal to the variance vector for the jth state of M_ϕ.

5. The articulatory-to-acoustic mapping $\boldsymbol{W}_k$ and its pseudo-inverse $\boldsymbol{W}_k^\dagger$ are appended to the model $\mathcal{M}_\phi$.

The mid-point and slope vectors of the above initial monophone MSHMMs are then re-estimated based on Equations (4.28) and (4.29) using the Viterbi-type training algorithm, as described in Section 4.4.2 on page 76. In the acoustic domain, the differences between the transformed trajectory values and the real acoustic feature vectors are used to estimate the covariance matrices.

Triphone MSHMMs training

Given the 49 monophone MSHMMs obtained above, a set of triphone MSHMMs are created based on a simple ‘backoff’ scheme, which is shown in Figure 7.3. The key variable in this ‘backoff’ scheme is the n_{min} parameter, which is a pre-determined integer indicating the minimum number of occurrence of a triphone in the training text (converted to triphone format) required to create a distict CD model. For example, if a triphone “a-b+c” appears at least n_{min} times in the training data, it will be created in the triphone model set. Otherwise, the corresponding left-context biphone “a-b” is considered. Similarly, if the number of instances of the biphone “a-b” in the training set n_{bi} is at least n_{min}, then “a-b” will be used in the triphone model set. Otherwise, the corresponding monophone “b” is used instead.

Clearly, there is a tradeoff when choosing the value of n_{min}. On one hand, small values of n_{min} result in a large number of triphone models and more accurate modelling of contextual effects. On the other hand, the increased number of model parameters due to a small value of n_{min} can always lead to data sparsity problems when estimating model parameters. Moreover, the consequent computation for a large number of models is more expensive. A study on the effects of different values of n_{min} for phone classification performance is presented in (Russell & Jackson 2005), which empirically demonstrates that $n_{min} = 50$ provides necessary accuracy and efficiency. In the research presented in this thesis, n_{min} is also set

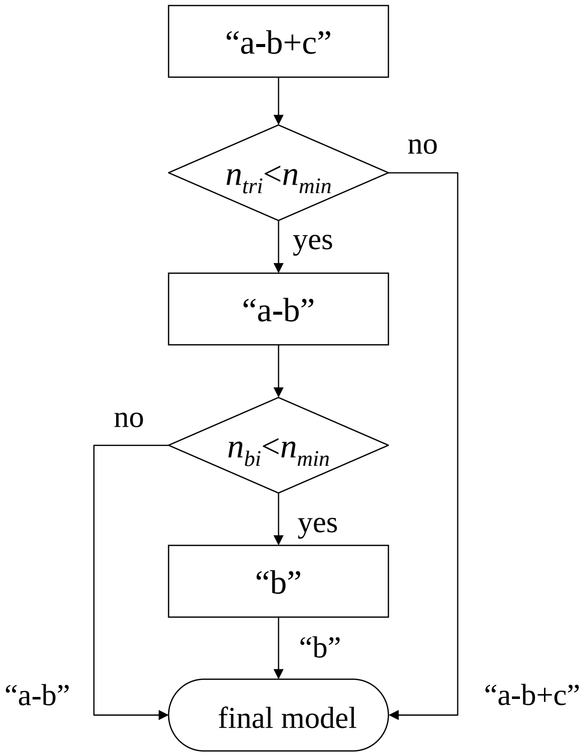


Figure 7.3: The 'backoff' scheme for triphone MSHMMs construction.

to 50, giving a set of 926 CD MSHMMs for the gender-dependent male system, and 1230 CD MSHMMs for the gender-independent system. The triphone MSHMMs are constructed by simply cloning monophones and their model parameters are then re-estimated using the same algorithm as in the case of monophone MSHMM training.

7.3.4 Non-linear/linear MSHMMs

Once a set of linear/linear MSHMMs has been built (either CI or CD), the speech parameter generation algorithm (see Section 5.2) can be applied to generate smoothed non-linear trajectories which then replace the piecewise linear trajectories in the intermediate layer of linear/linear MSHMMs. As has been justified in Section 5.2, among the three alternative SPG algorithms, the first algorithm is the key one that actually does the non-linearisation work and the other two algorithms are extensions of the first SPG algorithm. Since the objective of this research is to smooth and non-linearize the piecewise linear trajectories in MSHMMs, the first speech parameter generation algorithm is most relevant and therefore is chosen for this research.

Complexity of the SPG algorithms

For N-best list rescoring, in principle, the three SPG algorithms are all valid, though each having different computational complexity. For the first SPG algorithm in which the state sequence is given, $\mathrm{O}(T^3D^3)$ operations are need to solve Equation 5.15 (on page 89), where T and D denote the length of the speech observation sequence and the dimensionality of the static observation vector, respectively. Tokuda et al. (2000) showed that this complexity can be reduced to $\mathrm{O}(T^3D)$ when a diagonal covariance is assumed, and it can be reduced further to $\mathrm{O}(TDL^2)$, where L denote the delta window size, by using the Cholesky decomposition or the QR decomposition.

In the second SPG algorithm, the state sequence is unknown. A recursion similar to the RLS algorithm was described in (Tokuda, Kobayashi & Imai 1995) to search for the optimal state sequence, which has time complexity of $\mathrm{O}(T^2D)$ if diagonal covariances are assumed. The third SPG algorithm is derived based on the EM algorithm, for which the time complexity is known as $\mathrm{O}(N^2T)$ in the general case, where N is the number of states in the HMM. Of course, the complexity of the first algorithm should be taken into consideration when analyzing the overall complexity for both the second and third SPG algorithm because the first algorithm is integrated into the other two algorithms.

Complexity is certainly an important factor when choosing an appropriate algorithm, and the first SPG algorithm obviously has advantages in this respect compared with the other two. However, the main reason that the first SPG algorithm is chosen for this research is that it focuses right on the non-linearisation of the piecewise constant trajectories, which suits the context of this research very well. The other two algorithms are less attractive because they involve heavy computation in, for example, searching for the optimal state space, which is not the focus of this work and could be done by some other means.

Validation of the SPG algorithm implementation

The first SPG algorithm is used in this research, which is implemented using Matlab (see Appendix D). Before the actual rescoring experiments start, it is necessary to validate this implementation to confirm that it fulfills the requirements of this research. The most important requirement is, obviously, that this algorithm can generate an appropriate non-linear trajectory. An objective evidence or measure would be most credible, though it is difficult to define the 'appropriateness' of a non-linear trajectory.

The method used in the validation experiments presented here is to predict the shape of the generated trajectory under certain conditions/assumptions based on the knowledge about this algorithm, and then compare the predicted trajectory with the one which is actually generated using the software implementation. If the actually generated trajectory coincide with what is expected, then we have confidence that the particular software implementation meets our requirements.

In the SPG algorithm, two type of constraints are used to ensure smoothed trajectories are generated. These can be seen from the following equation, which is used to find the solution for the non-linear trajectory $\boldsymbol{C}$

$$\boldsymbol{W}^{\top}\Sigma_x^{-1}\boldsymbol{W}\boldsymbol{C} = \boldsymbol{W}^{\top}\Sigma_x^{-1}\boldsymbol{M}^{\top}, \tag{7.7}$$

where the definition of the symbols can be found in Section 5.2 on page 87. The first and also the most important constraint is given by $\boldsymbol{W}$, which is defined as $\boldsymbol{O} = \boldsymbol{W}\boldsymbol{C}$, i.e., the matrix which transform static feature space into a larger space which includes both static and dynamic features. Without this constraint, the generated trajectory $\boldsymbol{C}$ will be a piecewise constant mean vector sequence $\boldsymbol{M}$ of the HMMs, as can be seen from Equation (7.7). The second constraint is integrated into the term Σ_x in Equation (7.7), i.e., the variances of the model. The variance determines how far the generated trajectory lies from the mean. When the variance is very small, for instance, it is expected that the generated trajectory will be very close to the piecewise mean vectors.

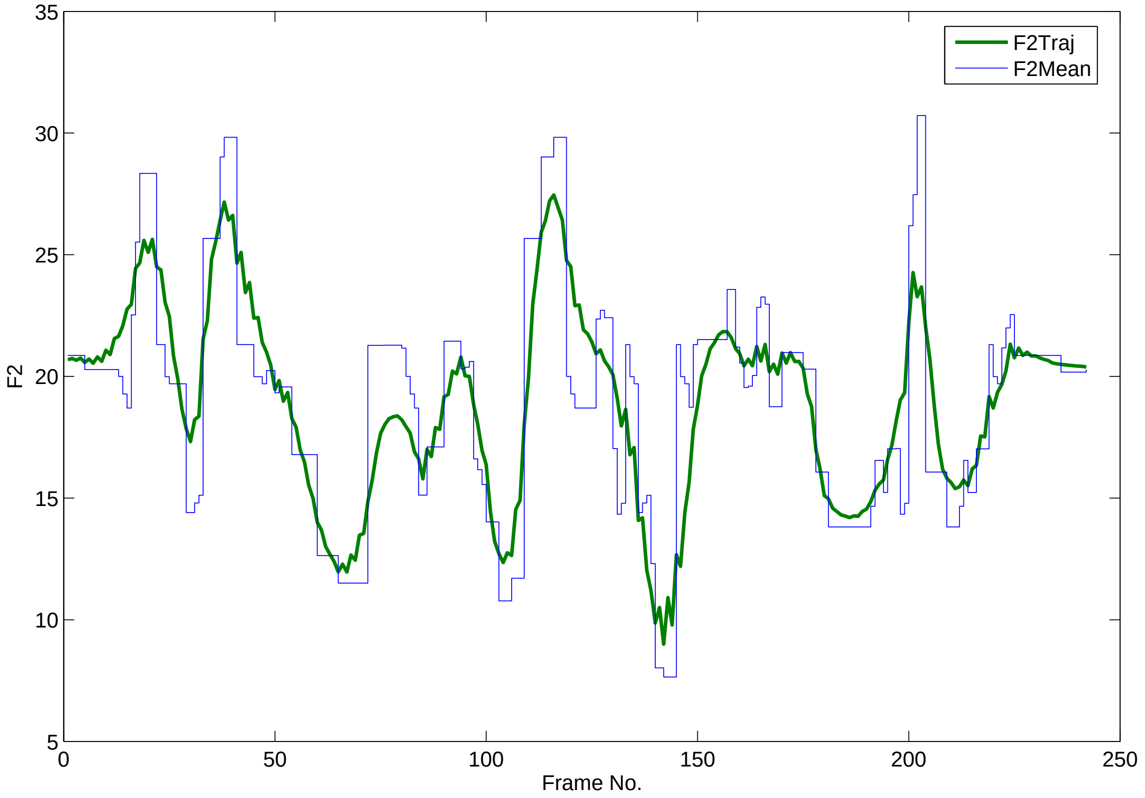


Figure 7.4: Formant trajectories for F2 generated from HMMs using the SPG algorithm implemented in this research. 'F2Mean' and 'F2Traj' are trajectories which are generated without and with the constraints between static and dynamic features, respectively.

Based on the above discussions, we can test the software by changing some parameters in $\boldsymbol{W}$ or Σ_x such that we can predict what the generated trajectory is going to look like. Then we can validate the algorithm by comparing the true generated trajectory with the predicted one. In doing so, firstly, the elements in the transform matrix $\boldsymbol{W}$ which correspond to the Δ and Δ^2 parameters are forced to be zero. This in effect disables the constraints of Δ and Δ^2 parameters, and therefore the generated trajectory should become a sequence of piecewise constant mean vectors. On the other hand, in the case that the Δ and Δ^2 constraints are applied, the generated trajectory is expected to be non-linear and smooth.

Figure 7.4 shows the generated formant trajectories for F_2 from HMMs by using the SPG algorithm. The thin line, denoted by 'F2Mean' in the plot, is generated by setting the parameters in $\boldsymbol{W}$ which correspond to the delta and delta-delta coefficients to be zero. As predicted, this line (i.e., 'F2Mean') coincides exactly with the piecewise constant mean vector sequence. When the constraint $\boldsymbol{O} = \boldsymbol{W}\boldsymbol{C}$ is imposed, the piecewise constant trajectories are smoothed to give a non-linear continuous trajectory, as shown by the thick line (denoted by 'F2Traj') in Figure 7.4. In addition, Figure 7.4 shows the phenomenon of 'target undershoot' which is often seen in formant synthesis, where a particular target (or formant) is not achieved due to the short duration of that target.

The next set of validation experiments concentrate on predicting the F_2 trajectory based on a single model 'aa' to gain a closer look at the behavior of the SPG algorithm. The following validation test conditions are considered:

1. $\Delta = 0$ and $\Delta^2 = 0$: under this condition, i.e., no constraint between static and dynamic features, the generated trajectory is expected to be the model means.

2. $\Delta \neq 0$ and $\Delta^2 = 0$: under this condition, the constraint of the Δ parameters is applied but there is no constraint on the Δ parameters (because $\Delta^2 = 0$). It is expected that the generated trajectory is smooth but not as smooth as that generated with both Δ and Δ^2 constraints.

3. $\Delta \neq 0$ and $\Delta^2 = 0$ and all variances are set at a small value 0.001: due to the small variances, the generated trajectory is expected to be very close to the means of the model.

4. $\Delta \neq 0$ and $\Delta^2 \neq 0$: both the Δ and Δ^2 constraints are applied so that the generated trajectory is expected to be the most smoothed among others.

5. $\Delta \neq 0$ and $\Delta^2 \neq 0$ and all variances are set at a small value 0.001: due to the small variances, the generated trajectory is expected to move towards the means of the model.

The corresponding trajectories generated using the SPG algorithm under these conditions are plotted in pair and shown in Figure 7.5. The trajectories F2A to F2E correspond to validation conditions 1 to 5. F2A is presented in every subplot for comparison, since it is

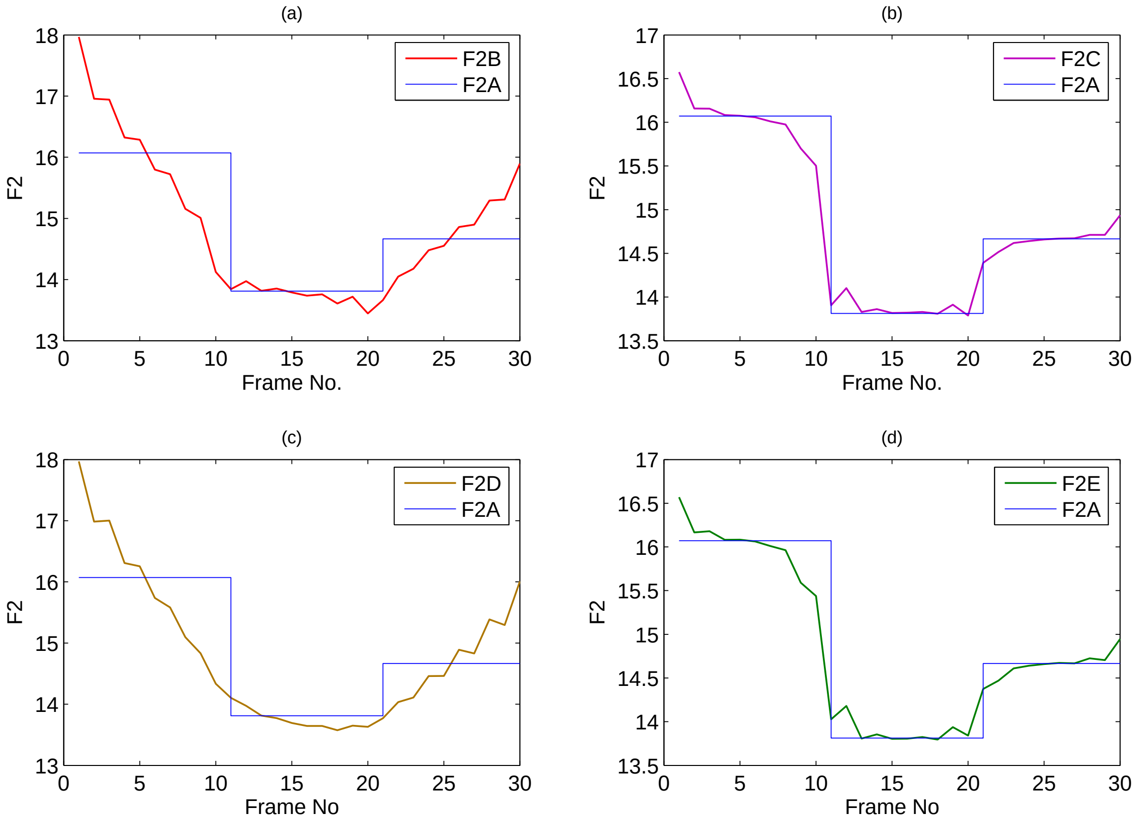


Figure 7.5: Formant trajectories for F2 generated from the HMM of the phone ‘aa’ by using the SPG algorithm implemented in this research. In the legend, F2A corresponds to the trajectory generated under the condition $\Delta = 0$ and $\Delta^2 = 0$. Similarly, F2B: $\Delta \neq 0$ and $\Delta^2 = 0$; F2C: $\Delta \neq 0$ and $\Delta^2 = 0$ and all variances at 0.001; F2D: $\Delta \neq 0$ and $\Delta^2 \neq 0$; F2E: $\Delta \neq 0$ and $\Delta^2 \neq 0$ and all variances at 0.001.

exactly the same with the model means (the RMSE between F2A and model means is 0, as shown in Table 7.3). As can be seen in Figure 7.5, the generated trajectories coincide with the predictions. In particular, subplot (b) and (d) look very similar, though in (b) F2C is generated without the Δ^2 constraint, and F2E in (d) is constrained by both the Δ and Δ^2 parameters. The small variances artificially used in both (b) and (d) ‘squeeze’ the trajectories F2C and F2E towards the model mean, resulting in small RMSE scores between these two generated trajectories and the models mean, as can be seen in Table 7.3. Also because of this reason, the effect of the Δ^2 constraint is less obvious in subplot (b) and (d). The trajectory F2D in subplot (c) is generated with both Δ and Δ^2 constraints, and therefore is smoother than F2B in (a) which is constrained only by Δ parameters. (Note F2B and F2D are both

	F2A	F2B	F2C	F2D	F2E
RMSE	0	0.017	1.0e-7	0.021	1.66e-7

Table 7.3: RMSE between the model mean and different trajectories (F2A-F2E) generated from the model 'aa' using the SPG algorithm implemented in this work under different validation conditions.

generated using the original variance of the model.) Reflected in Table 7.3, F2D has a slightly higher RMSE than F2B, which is 0.021 and 0.017, respectively.

The validation experiment results confirm that the software implementation of the SPG algorithm meets the requirements and can be used for the subsequent rescoring experiments. In order to be able to use the SPG algorithm, a set of conventional HMMs is built on the 12 PFS data, augmented with Δ and Δ^2 parameters, using HTK. The parameters of these PFS-based conventional HMMs are used to compute the non-linear trajectories. Given a state sequence (as described in the next section) and this 12 PFS-based model set, the SPG algorithm is applied to generate smooth formant trajectories. These non-linear trajectories will be used to rescore the N-best list against the piecewise linear trajectories as used in linear/linear MSHMMs.

7.4 Rescoring experiments

Having described the N-best list (Section 7.2) and the MSHMM model sets (Section 7.3), this section describes the rescoring experiments using the N-best list and MSHMMs. The description in this section is based on gender-dependent MSHMMs and the 1000-best list, which also applies to gender-independent MSHMMs rescoring using the 100-best list. The rescoring experiments can be divided into two categories: CI MSHMMs and CD MSHMMs rescoring experiments, as described in Sections 7.4.1 and 7.4.2, respectively. In both categories, rescoring the N-best list using linear/linear MSHMMs is carried out first as the baseline, followed by rescoring with non-linear/linear MSHMMs. In either category, it is expected that the use of non-linear formant trajectories will result in improved recognition performance compared with a piecewise linear trajectory model.

7.4.1 Monophone rescoring experiments

In monophone rescoring experiments, a set of 49 monophone MSHMMs corresponding to the TIMIT 49-phone set, as described in Section 7.3.2 and Section 7.3.3, is used to rescore the 1000-best list for each of the 128 test utterances.

Rescoring experiments with linear/linear MSHMMs

From Section 7.2.3 we know that, if a state sequence s corresponding to an N-best list hypothesis $\tilde{\mathcal{W}}_m$ is known, then Equation (7.6) can be used to calculate the probability $p(\boldsymbol{O}|\tilde{\mathcal{W}}_m)$ of the unknown test utterance $\boldsymbol{O}$ given the hypothesis (and MSHMMs). For linear/linear MSHMMs rescoring, the state sequence s can be obtained by using the following two different methods.

Constrained state sequences This type of state sequence is generated based on conventional HMMs. It has been explained in Section 7.2.2 that the 1000-best list is generated using a set of conventional triphone HMMs. For a given test utterance $\boldsymbol{O}$ and its corresponding 1000-best list, each 1000-best list hypothesis can be treated as the ‘transcription’ of the test utterance $\boldsymbol{O}$ (though the transcription may contain errors). Then the optimal state sequence between the conventional HMMs and the test utterance $\boldsymbol{O}$ is computed using ‘forced alignment’ implemented in the HTK decoder, taking account of the phone boundaries specified in the 1000-best list. The resulting state sequence is referred to as a ‘constrained’ state sequence. In the rescoring experiments, the monophone MSHMMs are forced to use this constrained state sequence to calculate probabilities.

Unconstrained state sequences The second type of state sequence used in the monophone MSHMMs rescoring experiments is the ‘unconstrained’ state sequence. The unconstrained state sequence is generated based on monophone MSHMMs. The Viterbi alignment between the monophone MSHMMs and the test utterance is performed using the ‘SEGVit’ toolkit. The term ‘unconstrained’ is used to indicate that the phone boundary timings of each 1000-best list hypothesis are ignored so that MSHMMs are free to choose duration length.

Both constrained and unconstrained state sequences are used to rescore the N-best list in linear/linear monophone MSHMMs rescoring experiments. However, only the constrained state sequence is used in the non-linear/linear MSHMMs rescoring experiments, as the non-linear trajectories are generated based on the constrained state sequences.

Rescoring experiments with non-linear/linear monophone MSHMMs

Rescoring the N-best list using non-linear/linear monophone MSHMMs is based on the same constrained state sequence as used in the linear/linear MSHMMs rescoring experiments and probabilities are calculated in the same way using Equation (7.6). Non-linear trajectories are first computed for each hypothesis in the 1000-best list based on the SPG algorithm, taking account of the constrained state sequence and model parameters of PFS-based conventional HMMs. When calculating probabilities using Equation (7.6) for linear/linear MSHMMs, the value of the piecewise linear formant trajectories at time t is determined by the corresponding MSHMM state occupied at time t. However, the value of the non-linear formant trajectories at time t is just a data point of the non-linear trajectories at that time.

7.4.2 Triphone rescoring experiments

The 926 triphone gener-dependent MSHMMs for male speakers (Section 7.3.3) are used to rescore the 1000-best list (Similarly, the 1230 CD gender-independent MSHMMs are used to rescore the 100-best list.). The rescoring process for triphone MSHMMs is similar to the monophone rescoring. In the triphone experiments, constrained state sequences are used.

Incompatibility between model sets

Before the triphone MSHMMs rescoring experiments start, the incompatibility problem between the model sets of the triphone MSHMMs and the conventional triphone HMMs must be solved. For monophone rescoring experiments, there is no such issue because both MSHMMs and standard HMMs use the same 49 symbols, so that both constrained and unconstrained state sequence can be used. However, the training of triphone MSHMMs is based on a ‘backoff’ scheme (see Section 7.3.3), while in HTK, decision-tree-based state-tying is employed to train a set of standard triphone HMMs. Therefore, the sizes of the two CD model sets (MSHMMs and HMMs) are different, with the triphone MSHMMs model set having 926 models and the standard triphone HMMs model set having 19802 models (before tying).

Due to the incompatibility problem between the model sets, the constrained state sequences generated using forced alignment based on the standard triphone HMMs model set for each 1000-best list hypothesis contain many models that do not exist in the MSHMMs triphone model set. A ‘backoff’-like method is used to map the unseen standard triphone HMMs to the 926 triphone MSHMMs. For each standard triphone HMM ‘a-b+c’, a search is made in the 926 triphone MSHMMs. If ‘a-b+c’ exists, the search stops and the next standard triphone HMM is selected. Otherwise, the left-biphone ‘a-b’ is searched in the 926 triphone MSHMMs. If ‘a-b’ exists, then the corresponding triphone MSHMMs for the standard triphone HMM ‘a-b+c’ is defined as ‘a-b’. If neither ‘a-b+c’ nor ‘a-b’ exist in the MSHMMs model set, then the corresponding triphone MSHMMs for the standard triphone HMM ‘a-b+c’ is chosen as the monophone ‘b’. The frequency counts for the backoff of ‘a-b+c’ → ‘a-b+c’, ‘a-b+c’ → ‘a-b’ and ‘a-b+c’ → ‘b’ are 364, 10714 and 8497, respectively.

Rescoring using linear/linear MSHMMs

The constrained state sequence for each 1000-best list hypothesis is generated in the same way as in the monophone rescoring experiment, based on a set of standard triphone HMMs. The state sequences are converted into such a format that any unseen standard HMM is replaced with its counterpart in the triphone MSHMMs model set, based on the model set mapping described in the previous section. The resulting state sequence is referred to as a transformed state sequence. The probability calculation is based on the transformed state sequence and the triphone linear/linear MSHMMs, which is calculated in the same way as in the monophone rescoring experiment.

Rescoring using non-linear/linear MSHMMs

Rescoring the 1000-best list using non-linear/linear triphone MSHMMs is based on the same state sequences used in the linear/linear triphone MSHMMs rescoring. However, the non-linear formant trajectories for each 1000-best list entry must first be generated. This involves training another set of standard triphone HMMs on formant data (12 PFS control parameters plus delta and delta-delta parameters). The PFS-based triphone HMMs model set is compatible with the MFCC-based triphone HMMs model set in that the same set of triphone symbols are used. For each 1000-best list hypothesis and it's corresponding state sequence, the speech parameter generation algorithm is applied using parameters of PFS-based triphone models to generate a non-linear formant trajectory.

For each 1000-best list hypothesis, the probability is calculated based on the transformed state sequence and the non-linear formant trajectories, as is the case of the monophone non-linear/linear MSHMMs rescoring in Section 7.4.1.

7.5 Rescoring experiment Results

This section presents experiment results for rescoring the N-best list. Monophone MSHMMs rescoring results are provided in Section 7.5.1, and triphone MSHMMs rescoring results are provided in Section 7.5.2. Section 7.5.3 gives a detailed analysis based on these experiment results.

7.5.1 Monophone rescoring results

Linear/linear MSHMMs rescoring results

Tables 7.4 and 7.5 show the rescoring results for linear/linear monophone MSHMMs. Note that constrained state sequences are used in experiments reported in Table 7.4, while unconstrained state sequences are used in experiments presented in Table 7.5. The bottom rows, denoted by Phone acc. (N+1), of both tables show the rescoring results after the correct transcriptions are appended to the 1000-best lists.

Mapping	1A	6B	10C	10D	49E
Phone accuracy	65.7	65.0	66.0	66.3	65.4
Phone acc.(N+1)	68.4	68.5	68.0	70.1	67.3

Table 7.4: Phone accuracy of rescoring using linear/linear monophone MSHMMs based on constrained state sequences on different mapping schemes. The five formant-to-acoustic mapping schemes are summarised in Table 4.1 on page 81.

Mapping	1A	6B	10C	10D	49E
Phone accuracy	65.3	65.8	66.1	66.4	66.4
Phone acc.(N+1)	67.8	68.4	69.7	68.6	70.4

Table 7.5: Phone accuracy of rescoring using linear/linear monophone MSHMMs based on unconstrained state sequences on different mapping schemes.

Table 7.4 shows that the MSHMMs based on mapping ‘10D’ give the highest phone accuracy, either without or with the reference transcriptions. However, Table 7.5 shows that the highest phone accuracy is achieved using ‘49E’ scheme when unconstrained state sequences are used. In the case when reference transcriptions are not included, the unconstrained rescoring results (Table 7.5) are generally better than the corresponding constrained results. Furthermore, the rescoring results based on unconstrained state sequences in Table 7.5 also show the trend that phone accuracy increases as the number of articulatory-to-acoustic mappings increases, as observed in (Russell et al. 2007).

Adding the correct transcript to the N-best list yields higher phone accuracies, as can be seen in Tables 7.4 and 7.5. When reference transcripts are included, the lowest phone accuracy (67.3%) occurs when constrained state sequences are used under mapping scheme 49E, as shown in Table 7.4, while the highest phone accuracy (70.4%) is also achieved using mapping 49E, though based on unconstrained state sequences (see Table 7.5). Even so, the lowest phone accuracy (67.3%) when reference transcripts are included is still higher than the highest phone accuracy (66.4%, unconstrained state sequence, 49E) when no reference transcripts are included.

The inclusion of correct transcripts in the N-best list is not unusual (Picone et al. 1999, Deng, Yu & Acero 2005). It provides an efficient means to find out how well the new acoustic models can perform. In an extreme case, the ideal models would give a 100%

phone accuracy in rescoring an N-best list which includes the reference transcript. In our rescoring experiments, it can be seen that when the correct transcrits are included in the N-best list, the monophone linear/linear MSHMMs would at best give a phone error rate close to 30%.

Non-linear/linear MSHMMs rescoring results

Mapping	1A	6B	10C	10D	49E
MSHMM(old)	66.1	66.4	66.4	66.4	66.7
Mapping reest.	66.2	66.7	66.7	66.9	67.0

Table 7.6: Phone accuracy of rescoring with non-linear/linear monophone MSHMMs using constrained state sequences on different mapping schemes. ‘Mapping reest.’ means MSHMM mappings are re-estimated.

Table 7.6 shows the results of rescoring the N-best list using non-linear/linear MSHMMs. MSHMM(old) means that the rescoring is based on the original MSHMM models, while ‘mapping reest.’ indicates that an updated MSHMM model set, whose articulatory-to-acoustic mapping parameters are re-estimated based on the non-linear formant trajectories, is used for rescoring (see below).

Mapping re-estimation in non-linear/linear MSHMMs The original mappings in linear/linear MSHMMs are trained based on the matched sequences of 12 PFS control parameters and 13 MFCCs on the TIMIT training set. These mappings remain intact in the rescoring experiments using linear/linear MSHMMs. However, in the non-linear/linear MSHMMs rescoring experiment, the role of the formant-to-acoustic mapping is to transform a *non-linear* formant trajectory, instead of a piecewise linear trajectory as in an linear/linear MSHMM, into the acoustic domain. Therefore, the original formant-to-acoustic mappings are not necessarily appropriate in the non-linear/linear MSHMM rescoring experiments. It might be beneficial if the formant-to-acoustic mapping is re-estimated based on matched sequences of non-linear formant trajectories in the 12PFS space and the 13-dimensional MFCC features. In doing so, for each utterance in the TIMIT training set, a non-linear trajectory is generated based on the speech parameter generation algorithm. Then the formant-to-acoustic mapping is re-estimated based on Equation (4.17), using matched sequences of non-linear 12 PFS-based trajectories and 13 MFCCs. As is shown in Table 7.6, the phone accuracies increase for all mapping schemes after the formant-to-acoustic mappings are re-estimated. This suggests that the re-estimated mapping provides a better fit between the acoustic features and the non-linear formant trajectories.

Because the non-linear/linear MSHMMs rescoring is based on the same constrained state sequence as in Table 7.4, the comparison should be made between Table 7.4 and Table 7.6. It can be seen that non-linear/linear MSHMMs outperform linear/linear MSHMMs for every mapping scheme, with at most a 4.9% reduction in phone error rate relative to linear/linear MSHMMs under mapping '6B'. The non-linear/linear MSHMM also outperforms the linear/linear MSHMM whose rescoring is based on unconstrained state sequences (Table 7.5). Table 7.6 also shows the trend that phone recognition performance increases as the number of mappings increases, with the highest phone accuracy 67.0% reached under the mapping scheme '49E' (i.e., the mapping scheme having 49 mappings).

7.5.2 Triphone MSHMMs rescoring results

Linear/linear MSHMMs rescoring results

Table 7.7 shows the triphone rescoring results using linear/linear MSHMMs. The mapping scheme '1A' gives the best phone accuracy 67.18% while mapping '6B' gives the lowest result 65.92%. The rescoring results for mapping schemes '10D' and '49E' are the same.

Mapping	1A	6B	10C	10D	49E
Phone accuracy	67.18	65.92	66.45	66.1	66.1

Table 7.7: Phone accuracy of rescoring with linear/linear triphone MSHMMs using constrained state sequences on different mapping schemes.

Non-Linear/linear MSHMMs rescoring results

Table 7.8 summarises the rescoring results using non-linear/linear triphone MSHMMs. The triphone non-linear/linear MSHMMs rescoring results follow a similar trend as the monophone non-linear/linear MSHMMs rescoring results, where the models whose formant-to-acoustic mappings are re-estimated give higher phone accuracies, except for the '49E' mapping scheme.

Mapping	1A	6B	10C	10D	49E
MSHMM(old)	66.45	66.49	66.56	66.45	66.62
Mapping reest.	66.54	66.52	66.60	66.60	66.47

Table 7.8: Phone accuracy of rescoring with non-linear/linear triphone MSHMMs using constrained state sequences on different mapping schemes. 'Mapping reest.' means MSHMM mappings are re-estimated.

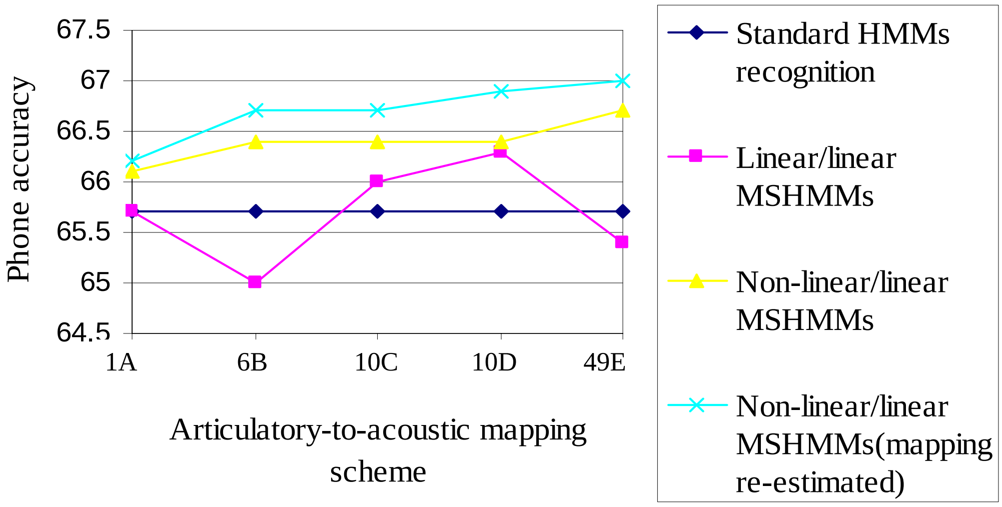
7.5.3 Rescoring results analysis

In order to better interpret the rescoring results, the monophone MSHMMs and triphone MSHMMs rescoring results are presented in Figure 7.6(a) and Figure 7.6(b), respectively, together with the baseline result 65.7%, which is obtained by using standard HMMs. Specific comments and detailed analysis on these results are presented in the following paragraphs.

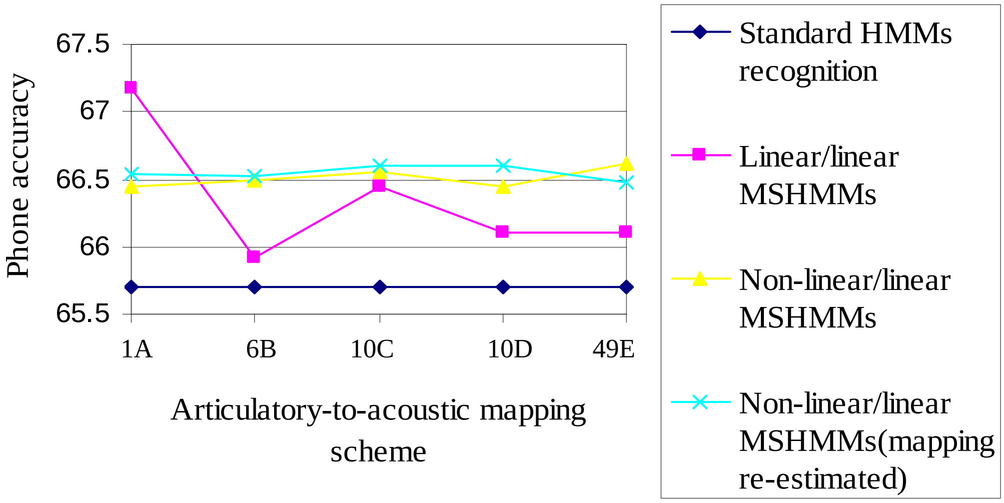
Statistical significance test

First of all, a paired t-test (see Section 6.3.3 on page 122) is again used to aid analysis of experimental results. As described before, the test set used in the N-best list rescoring experiments comprises 128 utterances from 16 male subjects in the TIMIT core test set, each subject speaking 8 sentences (excluding 'sa1' and 'sa2' sentences). These 128 test utterances are divided into 16 groups based on the identity of the speaker; that is, each group contains 8 utterances spoken by one of the 16 speakers.

Phone recognition accuracies are re-computed on a speaker-by-speaker basis. These new, speaker-dependent recognition results are then compiled and compared under a variety of scenarios (e.g., comparison of rescoring results between MSHMMs with piecewise linear trajectories and non-linear trajectories). The p-values for these paired t-test are summarised in Table 7.9. This table will be referred to several times in the following discussion.



(a)



(b)

Figure 7.6: (a) Monophone MSHMMs rescoring experiment results. (b) Triphone MSHMMs rescoring experiment results.

Mapping		1A	6B	10C	10D	49E
Monophone	Cons. vs. uncons.	0.27	$\underline{0.04}$	0.33	0.45	$\underline{0.02}$
MSHMMs	Linear vs. nonlinear	0.24	$\underline{0.01}$	0.13	0.19	$\underline{0.01}$
	Ori. vs. newMap	0.37	0.17	0.17	0.23	0.27
Triphone	Linear vs. nonlinear	0.06	0.34	0.36	0.07	0.16
MSHMMs	Ori. vs. newMap	0.21	0.44	0.36	0.13	0.15
Mono vs. Tri	Mono-linear vs. tri-linear	$\underline{0.02}$	$\underline{0.02}$	0.36	0.36	$\underline{0.02}$
	Mono-nonlinear vs. tri-linear	0.11	0.32	0.36	0.21	0.13
	Mono-nonlinear vs. tri-nonlinear	0.34	0.43	0.43	0.41	0.25

Table 7.9: Paired t-tests for rescoring results. This table shows the p-values for a variety of paired t-tests. Underlined p-values indicate that there are significant differences between test results with a conventional significance level $p < 0.05$. Abbreviations:
Cons.: constrained state sequence used in rescoring
Uncons.: unconstrained state sequence used in rescoring
Linear: piecewise linear formant trajectories used in MSHMMs
Nonlinear: nonlinear formant trajectories used in MSHMMs
Ori.: original formant-to-acoustic mappings used in rescoring
NewMap: formant-to-acoustic mappings re-estimated
Mono-linear & mono-nonlinear: monophone MSHMMs using piecewise linear trajectories or nonlinear trajectories
Tri-linear & tri-nonlinear: triphone MSHMMs using piecewise linear trajectories or nonlinear trajectories

Non-linear trajectory model v.s. linear trajectory model

Table 7.10 shows the *relative* improvement in phone accuracy that results from using non-linear formant trajectories instead of linear formant trajectories in the intermediate layer of both monophone MSHMMs and triphone MSHMMs, taking account of different formant-to-acoustic mapping schemes. Both monophone and triphone MSHMMs benefit from the use of non-linear formant trajectories in most cases. For example, the monophone rescoring results show that 4.9% and 4.6% reduction in phone error rates for the mapping schemes '6B' and '49E', repectively, are achieved by using non-linear formant trajecotries. Paired t-tests show that these two improvements are significant, as can be seen in Table 7.9, indicated by underlined p-values in the row of 'monophone MSHMMs–linear v.s. nonlinear'. However, an exception occurs in the triphone rescoring experiments when the mapping scheme '1A' is used, with the phone accuracy for linear/linear MSHMMs higher than the non-linear/linear MSHMMs (as can be seen in Figure 7.6(b)), though a paired t-test discovers that this difference is not significant.

Mapping	1A	6B	10C	10D	49E
Monophone	0.8%	**2.6%**	1.1%	0.9%	**2.4%**
Triphone	-1.0%	0.9%	0.2%	0.8%	0.6%

Table 7.10: Relative improvement in phone accuracy by using non-linear formant trajectories instead of linear formant trajectories in the intermediate layer of both monophone and triphone MSHMMs with regard to different articulatory-to-acoustic mapping schemes.

The comparison between rescoring results using linear and non-linear formant trajectories confirms our expectation that the use of the non-linear formant trajectories (generated by the speech parameter generation algorithm) in an MSHMM can result in improved recognition performance, relative to a piecewise linear trajectory model as used in linear/linear MSHMMs. Moreover, in some cases, e.g., in mappings ‘6B’ and ’49E’ for CI MSHMMs, this improvement is statistically significant.

As can be seen from Table 7.10, the performance gain (resulted from the use of non-linear formant trajectories in the intermediate layer of the MSHMM) for the monophone MSHMM system is greater than that for the triphone MSHMM system. This is also supported by the results of the statistical significance test (see the two rows in Table 7.9, ‘monophone MSHMMs–linear vs. nonlinear’ and ‘triphone MSHMMs–linear vs. nonlinear’), which show that except for monophone MSHMMs in ‘6B’ and ’49E’, the rest of MSHMMs, either monophone or triphone, do not demonstrate this statistical significance in performance improvement.

In Section 6.3.3, it was argued that the non-linear trajectory generated by the speech parameter generation algorithm might be capable of modelling coarticulation effects, evidenced by the results of the intelligibility and naturalness tests. The speech spectrogram (Figure 6.6 on page 125) shows smooth formant trajectories, which are able to match the actual formant trajectories in real speech quite well (see Figure 6.7 on page 127). In those speech synthesis experiments, only CI models were used. In the speech recognition experiments presented here, both CI and CD models are used. CD models (e.g., triphones) are designed to account for contextual effects, such as coarticulation, by taking into consideration the immediate preceding and following neighbours of a given phone. In practice, improved recognition performance can almost certainly be achieved when using triphone models instead of monophone models.

In the rescoring experiments reported here, however, contextual effects might have been characterised by: 1) nonlinear formant trajectories, or 2) CD MSHMMs. The different level of performance gain for monophone and triphone systems due to the use of non-linear trajectories seems to suggest that the use of the non-linear formant trajectories is a more effective means in modelling contextual effects than CD models for MSHMMs. To find out whether this is the case, rescoring results between the monophone and triphone systems are compared and analysed, as described below.

CI model v.s. CD model

It was expected that the use of triphone MSHMMs in the rescoring experiment would result in further recognition improvement compared to monophone MSHMMs. To find out to what extent the use of triphone MSHMMs improve the rescoring results, monophone and triphone rescoring results are compared. The comparison is divided into three categories: linear/linear MSHMMs rescoring, nonlinear monophone MSHMMs vs. linear triphone MSHMMs and non-linear/linear MSHMMs rescoring, as described below.

Linear/linear MSHMMs rescoring For linear/linear MSHMMs rescoring, in which piecewise linear trajectories are used in both monophone and triphone MSHMMs, it can be seen from Figure 7.7 that triphone MSHMMs outperform monophone MSHMMs for all mapping schemes but the ‘10D’ scheme. When the mapping scheme ‘1A’ is used, for instance, the triphone phone accuracy is 1.5% higher than the monophone phone accuracy. Rescoring based on other mapping schemes, such as ‘6B’,‘10C’ and ‘49E’, also benefit from the use of CD triphone models, though the improvement for these mapping schemes are all less than 1.5%.

Paired t-test results (Table 7.9, row ‘mono vs. tri–mono-linear vs. tri-linear’) show that the use CD models instead of CI models results in statistically significant improvements for

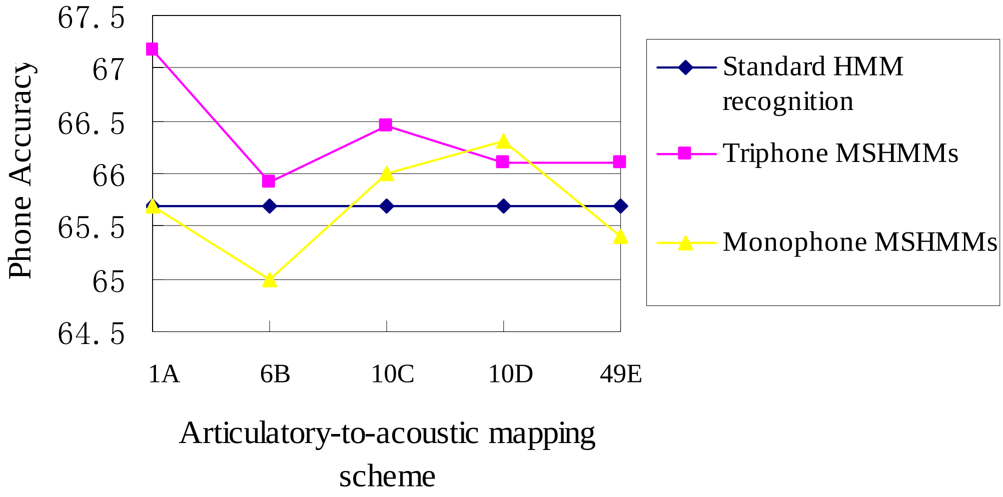


Figure 7.7: Comparison between monophone and triphone MSHMMs rescoring results using linear/linear MSHMMs.

mappings 1A, 6B and 49E, but not for 10C and 10D. There is no surprise that the triphone system outperforms the monophone system because triphone MSHMMs contain contextual information that CI monophone MSHMMs do not. However, what happens if the piecewise linear trajectories in the monophone MSHMMs are replaced by non-linear trajectories, while triphone MSHMMs still use piecewise linear trajectories? Do CD linear/linear MSHMMs still outperform ‘upgraded’ nonlinear/linear monophone MSHMMs? This is discussed in the next set of comparison.

Nonlinear trajectory monophone MSHMMs vs. linear trajectory triphone MSHMMs

We now compare rescoring results between nonlinear/linear CI monophone MSHMMs (mapping re-estimated) and linear/linear CD triphone MSHMMs. The only difference between this comparison and the comparison described in the previous paragraph is that the monophone models now contain some contextual information due to the use of smoothed, nonlinear formant trajectories in the intermediate layer.

As can be seen in Table 7.11, nonlinear trajectory monophone MSHMMs outperform linear trajectory triphone MSHMMs for all mapping schemes but 1A, though paired t-tests

Mapping	1A	6B	10C	10D	49E
Monophone	66.2%	66.7%	66.7%	66.9%	67%
Triphone	67.18%	65.92%	66.45%	66.1%	66.1%

Table 7.11: Comparison of rescoring results between nonlinear monophone MSHMMs (mapping re-estimated) and linear triphone MSHMMs.

(Table 7.9, row 'mono vs. tri–mono-nonlinear vs. tri-linear') show that there are no significant differences between monophone and triphone results for all mapping schemes. In other words, monophone MSHMMs using nonlinear formant trajectories can produce similar level of performance as that of triphone MSHMMs using piecewise linear trajectories.

Based on the analysis of rescoring results between CI MSHMMs and CD MSHMMs so far, we learn that

- when piecewise linear trajectories are used in both monophone and triphone MSHMMs, CD triphone MSHMMs perform consistently better than CI monophone MSHMMs.
- when nonlinear trajectories are used in monophone MSHMMs and piecewise linear trajectories are retained in triphone MSHMMs, there are no significant differences between CI and CD MSHMMs rescoring results.

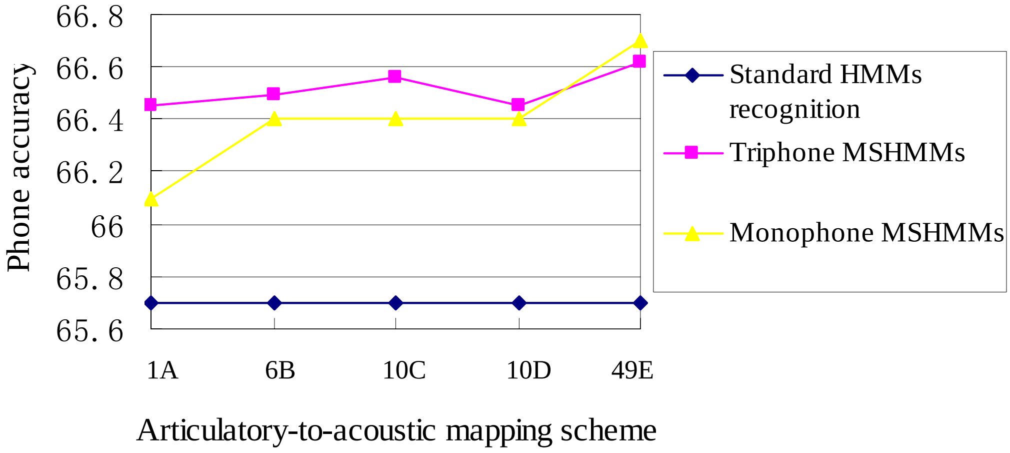
The combined analysis implies that *the smoothing of non-linear formant trajectories in the intermediate layer of an MSHMM is capturing the contextual effects that CD MSHMMs are designed to accommodate.* This is due to the use of the speech parameter generation algorithm. Obviously, the main advantage of a monophone system over a triphone system is fewer models and parameters, hence faster training and decoding. The main disadvantage of conventional CI models is that they are less accurate than CD models because no contextual information is taken into account. However, the use of non-linear formant trajectories in CI MSHMMs provides a means of contextual modelling, while retaining the advantages of parameterisation and computation of CI models. A practical disadvantage of the non-linear trajectory approach is that the generation of the nonlinear trajectories is computationally expensive.

Non-linear/linear MSHMMs rescoring The last set of comparison between monophone and triphone MSHMMs rescoring results focuses on the use of the nonlinear formant trajectories, as shown in Figure 7.8. In these experiments, non-linear formant trajectories, rather than piecewise linear trajectories, are used in both monophone and triphone MSHMMs.

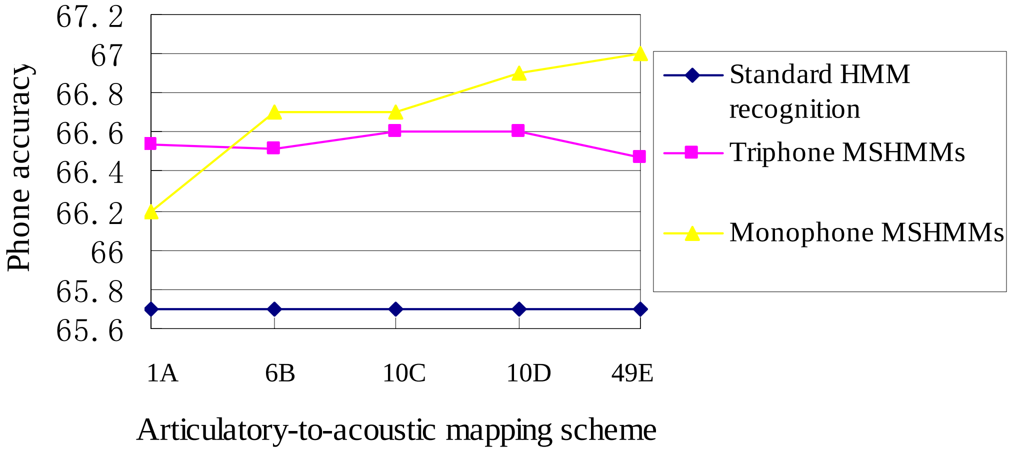
Figure 7.8(a) compares rescoring results using non-linear formant trajectories in both monophone and triphone models without re-estimating the formant-to-acoustic mapping. It can be seen that the triphone rescoring results outperform monophone rescoring results for the four mapping schemes '1A, 6B, 10C, 10D', though the improvement is modest. For the mapping scheme '49E', monophone MSHMMs performs slightly better than triphone HMMs. However, the comparison between rescoring results using non-linear/linear MSHMMs for monophone and triphone models with re-estimated articulatory-to-acoustic mappings is a different story, as shown in Figure 7.8(b). It can be seen that, except for the mapping scheme '1A', monophone MSHMMs perform better than triphone MSHMMs after the formant-to-acoustic mappings are re-estimated.

Paired t-tests show that none of these differences between monophone and triphone MSHMMs rescoring results based on nonlinear formant trajectories is statistical significance, with or without formant-to-acoustic mappings re-estimated (Table 7.9, row 'mono vs. tri– mono-nonlinear vs. tri-nonlinear' gives the paired t-test results when formant-to-acoustic mappings are re-estimated).

In summary, the comparison between monophone and triphone MSHMMs rescoring results shows that, in the case that non-linear formant trajectories are used in both monophone and triphone MSHMMs, the use of CD models does not improve recognition performance over CI MSHMMs. Based on the above analysis, one possible explanation is that nonlinear smooth formant trajectories accommodate contextual effects so that the use of CD triphone models has less effects.



(a)



(b)

Figure 7.8: Comparison between monophone and triphone MSHMMs rescoring results using non-linear formant trajectories in the intermediate layer. (a) The articulatory-to-acoustic mappings are not updated. (b) The articulatory-to-acoustic mappings are re-estimated based on the non-linear formant trajectories of the training utterances.

Advantages of formant-to-acoustic mapping re-estimation

Table 7.12 shows relative improvements due to the re-estimation of the formant-to-acoustic mapping for both monophone MSHMMs and triphone MSHMMs. When nonlinear formant trajectories are used, both monophone and triphone MSHMMs benefit from the re-estimation of the articulatory-to-acoustic mappings. The re-estimated mapping is computed using matched sequences of acoustic data and nonlinear trajectory formant data. As a result, the re-estimated mapping provides a better fit between the acoustic data and nonlinear formant trajectories. Note that the use of the re-estimated mappings in the '49E' scheme of the triphone experiment yields a slight lower phone accuracy. However, paired t-tests (see Table 7.9, rows 'monophone MSHMMs–ori. vs. newMap' and 'triphone MSHMMs–ori. vs. newMap') reveal that none of these differences in recognition performance caused by mapping re-estimation is statistical significance.

Mapping	1A	6B	10C	10D	49E
Triphones	0.1%	$<0.1\%$	$<0.1\%$	0.2%	-0.2%
Monophones	0.2%	0.5%	0.5%	**0.8%**	0.4%

Table 7.12: Relative improvement in phone accuracy by using articulatory-to-acoustic mapping re-estimation for monophone and triphone MSHMMs rescoring experiment.

Although investigating alternative formant-to-acoustic mapping is out of the scope of this thesis, it should be noted that linear mappings are not appropriate to model the relationship between formant-based and acoustic representations of speech, which is non-linear per se (Russell et al. 2007). Even with linear mappings, the use of articulatory-to-acoustic mappings that are 'pre-trained' on matched data is unlikely to be optimal (Russell & Jackson 2005).

Comparison between constrained and unconstrained state sequences

In CI linear/linear MSHMMs rescoring experiments, both constrained and unconstrained state sequences are used, as described in Section 7.4.1 on page 156. The main difference between constrained and unconstrained state sequences is that the generation of constrained state sequences uses phone boundaries based on standard HMMs, while the generation of unconstrained state sequences is based on MSHMMs, ignoring information about phone boundaries.

Mapping	1A	6B	10C	10D	49E
Constrained	65.7%	65.0%	66.0%	66.3%	65.4%
Unconstrained	65.3%	65.8%	66.1%	66.4%	66.4%

Table 7.13: Comparison of CI linear/linear MSHMMs rescoring results using constrained and unconstrained state sequences.

The rescoring results using CI linear/linear MSHMMs based on constrained and unconstrained state sequences are shown in Table 7.13. As can be seen, unconstrained state sequences give higher phone accuracies than constrained state sequences for all mapping schemes but 1A. A possible reason is that performance for mapping scheme 1A, which has a single mapping for all phone class, is less reliable than other phone-class-dependent mapping schemes (see the discussion below on different mapping schemes). Paired t-tests (Table 7.9, row 'monophone MSHMMs–cons. vs. uncons.') show that differences between phone accuracies for mapping schemes 6B and 49E are statistically significant.

In generating unconstrained state sequences, no phone boundary timings are specified in the transcripts. That is to say, only phone labels are provided when performing Viterbi alignment. In this case, the MSHMMs are free to choose any legitimate durations. The improvement in rescoring results resulting from the use of unconstrained state sequences demonstrate the advantage of the improved duration modelling in the MSHMM framework, relative to standard HMMs.

Comparison between different mapping schemes

An important conclusion drawn from previous work on linear/linear MSHMMs is that phone classification and recognition performance can be improved by increasing the 'richness' of the formant-to-acoustic mapping, and the most reliable mapping scheme which gives the best performance is 49E (Russell & Jackson 2005, Russell et al. 2007). In the 49E mapping scheme, a total of 49 mappings are used, and each of the 49 phone-level models has a separate formant-to-acoustic mapping, which is estimated based on the match sequences of

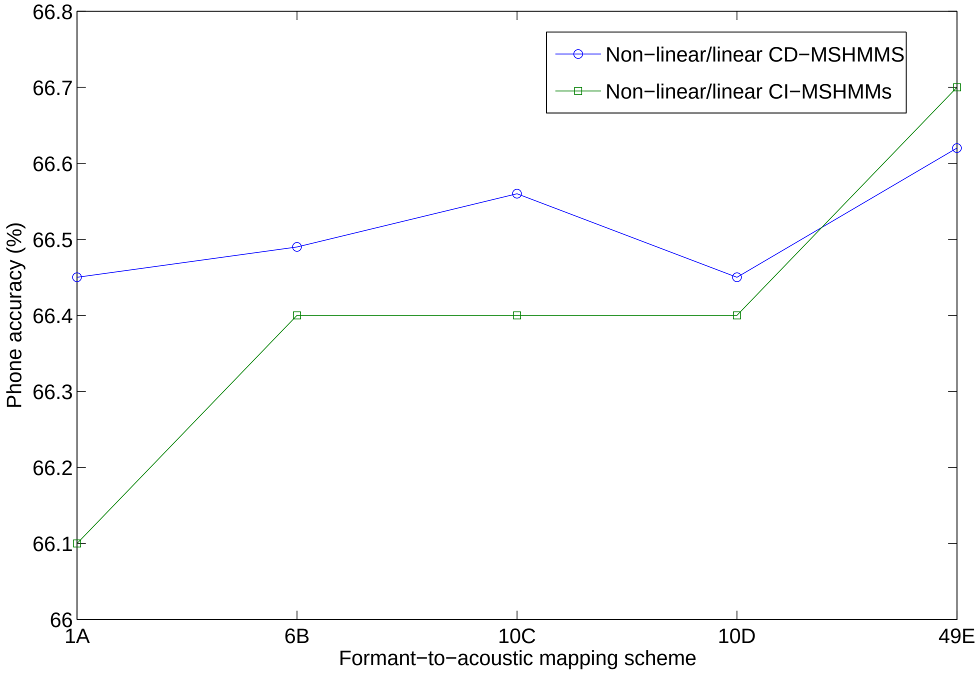


Figure 7.9: The effect of different formant-to-acoustic mapping schemes on recognition performance for non-linear/linear MSHMMs.

formant and acoustic vectors corresponding to that phone (see Table 4.1 on page 81).

For non-linear/linear MSHMMs, it is worthwhile to find out whether the above linear/linear MSHMM conclusion on different mapping schemes is still applicable. Figure 7.9 shows the effect of different formant-to-acoustic mapping schemes (A-E) on recognition performance for non-linear/linear MSHMMs. Firstly, for both non-linear/linear CI and CD MSHMMs, the best performance is indeed achieved when the mapping scheme 49E is used, and the mapping scheme 1A gives worst performance in either case. These results confirm the reliability of the mapping scheme 49E, as in the case of the linear/linear MSHMMs, and demonstrate the limitation of using a single linear mapping for all phone class (i.e., 1A). Provided that there are sufficient training data, ideally, we want to use as many as possible formant-to-acoustic mappings, each tailored to an individual model, to achieve best fit between formant and acoustic data.

Secondly, the differences in performance among mapping schemes 6B, 10C and 10D are not significant. In fact, for non-linear/linear CI MSHMMs, mapping schemes 6B, 10C and

10D give virtually the same phone accuracy 66.4%. For CD MSHMMs, the phone accuracy increases from 6B to 10C, and then drops slightly when 10D is used, though in all cases, the phone accuracy is around 66.5%. Similar tendency of performance for these three mapping schemes can be also found in previous results on linear/linear MSHMMs (see Russell & Jackson (2005)). A direct reason could possibly be that the numbers of distinct mappings for these three mapping schemes are 6, 10 and 10, respectively, not very different from each other. Moreover, some of the mappings (e.g., those for vowels and nasal, see Table 4.1 on page 81) are actually same across these three mapping schemes. Therefore, it is not suprising that these three mapping schemes (6B, 10C and 10D) give similar level of performance.

Finally, the motivation for the use of phone-class-dependent mapping schemes 6B, 10C and 10D is described in (Russell & Jackson 2005). For example, categorisation 6B is motivated by different linguistic classes, while categorisation 10D is motivated by a classification of speech production mechanisms into discrete features based on source-filter theory. The scheme 10C is proposed by Deng & Ma (2000), which is based on different manner of articulation. Although each of these three categorisations has sound theoretical motivation, the performance may be compromised by the limitation of the *linear* mappings to transform knowledge at production level onto acoustic space, which is certainly a invalid assumption since the relationship between formant and acoustic representations is not linear (Richards & Bridle 1999, Russell et al. 2007). Re-estimating formant-to-acoustic mappings based on matched non-linear formant trajectories and acoustic vector sequences, as described in Section 7.5.3 on page 172, might be helpful in improving the match between these two speech representations and hence recognition performance. However, this is an ad hoc rather than essential solution since the linearity of the formant-to-acoustic mapping is unchanged.

Comparison to the standard HMMs recognition result

The standard HMMs recognition result is shown in all figures presented in this section as a baseline result. It can be seen from Figure 7.6 that all triphone rescoring results are higher than the standard HMMs recognition result. The monophone rescoring results for non-linear/linear MSHMMs exceed the standard HMMs result, while for linear/linear monophone MSHMMs, only the mapping schemes ‘10C’ and ‘10D’ give higher phone accuracies than standard HMMs, with ‘1A’ giving exactly the same result as standard HMMs.

One-sample t-tests are performed to assess the statistical significance between the mean values of various MSHMMs rescoring results and the baseline result of standard HMMs. However, the t-test results discover that none of these differences is statistically significant. Further development on MSHMMs is required if the ultimate goal is to achieve improved performance compared with state-of-the-art HMM-based ASR systems.

TIMIT core test set experiment results

Up until now, research on MSHMMs have been exclusively based on gender-dependent models (Russell & Jackson 2005, Russell et al. 2007). The motivation for using gender-dependent models rather than gender-independent models in MSHMMs is described in Section 3.3 on page 66, where the speech data (i.e., TIMIT) used for this research is introduced. Since this thesis work builds on the linear/linear MSHMM described in (Russell & Jackson 2005, Russell et al. 2007), it is natural that the emphasis of this research is also placed on the gender-dependent MSHMMs to demonstrate the effect of introducing a non-linear formant trajectory model. Therefore, the discussions and analysis on recognition experiments reported in this chapter have by far concentrated on rescoring the 1000-best list using the gender-dependent MSHMMs for male speakers.

In the speech recognition research community, however, a well-known benchmark test is phonetic recognition on the TIMIT core test set (which contains 24 speakers, 2 male and 1 female from each of the 8 dialect region, giving a total of 192 test utterances). For comparison, a second N-best list is also used in the recognition experiments, which is a 100-best list generated based on the TIMIT core test set (see Section 7.2.2), in addition to the 1000-best list for male speakers. Experiment results on this standard test set can be appreciated in a broader context of this field, allowing us to measure the distance in performance between our system and those best performing systems in the field.

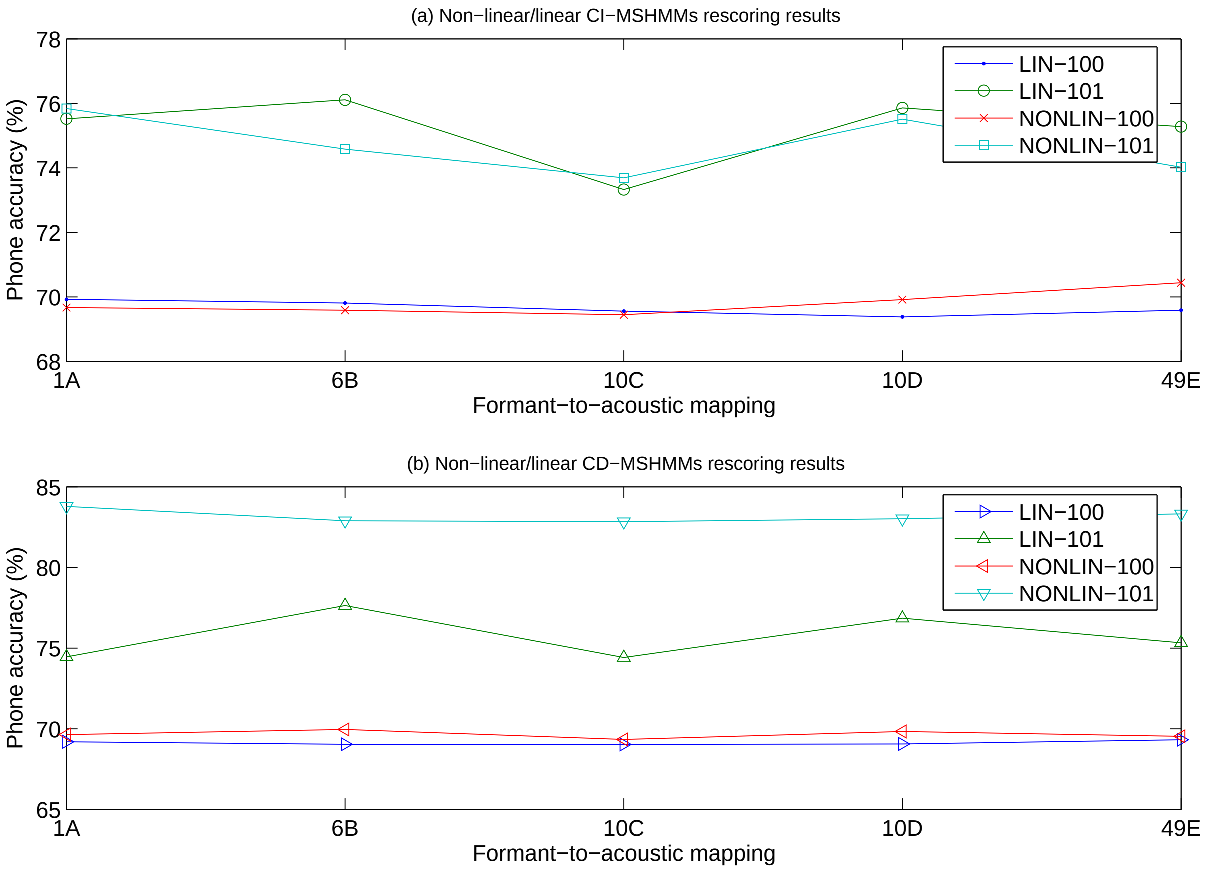


Figure 7.10: 100-best list rescoring results on the TIMIT core test set using gender-independent MSHMMs. (a) CI MSHMMs rescoring results. (b) CD MSHMMs rescoring results. In the legend, LIN and NONLIN denote linear trajectory or non-linear trajectory model, respectively, and '101' means the reference label file are included in the 100-best list.

Rescoring the 100-best list on the TIMIT core test set requires a set of gender-independent MSHMMs built on the full training set of the TIMIT database (using data for both male and female speakers), as described in Section 7.3. As has been stated before, this is the first time in the context of the MSHMM that gender-independent models are used. Figure 7.10 summarises results for resocring the 100-best list using gender-independent MSHMMs on the TIMIT core test set. A variety of model sets are considered, including CI and CD models, linear/linear MSHMMs and non-linear/linear MSHMMs (denoted by LIN and NONLIN respectively), and all five formant-to-acoustic mapping schemes (1A-49E). In addition to the 100-best list rescoring results, the corresponding 101-best list results (indicated by '101') are also presented in Figure 7.10. The 101-best list is generated by appending the reference label file to the original 100-best list.

The TIMIT core test set experiment results show that non-linear/linear MSHMMs outperform linear/linear MSHMMs in most cases, as can be seen in Figure 7.10. Statistical significance tests are also performed to aid the analysis of recognition results. When the reference label files are not included in the N-best list, paired t-test results reveal that significant improvements in recognition performance are achieved under mapping schemes 49E for CI MSHMMs, and 6B and 10D for CD MSHMMs. In the case that the reference label files are also included in the N-best list, significant improvements are obtained for non-linear/linear CD MSHMMs for all mapping schemes, with at most 46% reduction in PER relative to linear/linear CD MSHMMs under the mapping scheme 49E. These results clearly demonstrate the advantage of adopting a non-linear trajectory model in the intermediate layer of MSHMMs.

Method (Author)	Phone accuracy (%)
CD triphone HMMs (Lamel & Gauvain 1993)	69.1
CI hybrid HMM/ANN (Robinson 1994)	73.9
Bayesian triphone HMMs (Ming & Smith 1998)	74.4
Anti-phone, heterogeneous classifiers (Halberstadt & Glass 1998)	75.6
HTMs (Deng & Yu 2007)	75.2
Augmented conditional random fields (Hifny & Renals 2009)	73.4
Deep belief networks (Mohamed, Dahl & Hinton 2012)	79.3

Table 7.14: Some of the best phone recognition results on the TIMIT core test set in the literature.

Some of the best phone recognition results on the TIMIT core test are provided in Table 7.14. The best published phone accuracy on the TIMIT core test set so far is 79.3% based on deep belief networks (Mohamed et al. 2012). In this work, the best result obtained on the same test set is 70.4% (in the case of non-linear/linear CI-MSHMMs under mapping 49E), which is higher than the result reported by (Lamel & Gauvain 1993) but lower than others. In addition, the best MSHMM result in this work is comparable to the baseline HMM system result on the TIMIT core test set, which is 70.6% phone accuracy, as can be seen in Table 7.1 on page 143. The results shown in Table 7.14 represent the state-of-the-art, which are clearly superior to the best result presented here. However, the focus of this work has been put on the *relative* performance of the non-linear trajectory model compared with the linear trajectory model. It is believed that a non-linear trajectory model of dynamics like the one developed in this study is essential in order to build a more advanced model which can compete with the state-of-the-art in the future.

7.6 Summary

This chapter presents an experimental study on using non-linear smooth formant trajectories in place of piecewise linear trajectories to model speech dynamics in the MSHMM framework. Tokuda's speech parameter generation algorithm is used to generate non-linear formant trajectories, and the resulting nonlinear/linear MSHMMs are evaluated and compared to linear/linear MSHMMs based on the N-best list rescoring paradigm.

Both CI and CD MSHMMs are used in the rescoring experiments to test the effectiveness of the nonlinear trajectory model of formant dynamics. Experiment results are reported and analysed, aided by statistical significance tests. Analysis of the results reveals that, in general, the use of a nonlinear trajectory model of dynamics in an MSHMM can lead to significantly improved recognition performance compared with an MSHMM using piecewise linear trajectories.

For gender-dependent MSHMMs, a careful comparison between CI and CD models is conducted which:

- confirms the superiority of CD MSHMMs over CI MSHMMs when piecewise linear trajectories are used in the intermediate layer in both cases,

- discovers that the smoothing in nonlinear formant trajectories is modelling contextual effects that CD models are designed to account for, evidenced by the similar level (in the sense of statistical significance) of recognition performance given by CI MSHMMs using nonlinear trajectories and CD MSHMMs using piecewise linear trajectories,

- finds out that there is no statistically significant difference between CI MSHMMs and CD MSHMMs rescoring results when nonlinear formant trajectories are used in both cases.

Linear/linear CI MSHMMs benefit from using unconstrained state sequences in rescoring (mappings 6B and 49E, in particular, give statistically significant improvements), while non-linear/linear MSHMMs benefit from re-estimation of formant-to-acoustic mappings (though not statistically significant). Inter-mapping comparison confirms the finding in (Russell et al. 2007) that best recognition performance is obtained by increasing the number of formant-to-acoustic mappings. Finally, a one-sample t-test shows that there is no statistical significance between the baseline HMM result and MSHMMs rescoring results.

Chapter 8

Conclusions

8.1 Individual contributions and main findings

The primary goal of the research presented in this thesis, as stated in Chapter 1, is to extend the work previously carried out by Russell et al. (Russell & Jackson 2005, Russell et al. 2007) to develop an improved, non-linear trajectory model of speech dynamics, relative to a piecewise linear trajectory model, within the framework of a multiple-level segmental HMM. This thesis describes an investigation into a particular type of non-linear trajectory model, in which smooth trajectories are generated by using Tokuda's speech parameter generation method. This study exploits the application of this non-linear trajectory model of formant dynamics to both speech synthesis and speech recognition.

8.1.1 Speech synthesis

The application of the speech parameter generation algorithm to HMM-based speech synthesis is certainly not original: see, for example, (Yoshimura99,Tokuda00,Tokuda02), in which the speech parameter generation algorithm is used directly in the acoustic domain to generate smoothed mel-cepstral parameters. A major contribution of the speech synthesis work presented in this thesis derives from the application of the speech parameter generation algorithm directly to a particular type of formant data: the 12-dimensional parallel formant synthesiser control parameters and the use of the classic HMS parallel formant synthesiser to generate speech.

In the context of an MSHMM, this study takes a first step towards the goal of a 'unified' model for speech recognition and synthesis. This research investigates the application of the speech parameter generation algorithm at a deeper, production level (i.e., the formant domain) to model speech dynamics. The particular type of intermediate representation concerned in this work comprises all 12 PFS control parameters, which, provided that these parameters are sufficiently accurate, can be used to generate high quality synthetic speech with a formant synthesiser. From this perspective, the novelty of the synthesis work presented in this thesis is that the proposed non-linear trajectory model is potentially suitable for both speech recognition and synthesis, and its synthesis performance is evaluated as another way of assessing the model.

In this research, the application of the speech parameter generation algorithm in the formant domain results in non-linear, smoothed formant trajectories. These continuous, non-linear formant trajectories have been shown to be able to account for the coarticulation and transitions between neighbouring phones, and match actual formant trajectories in real human speech quite closely. This is advantageous over a piecewise linear trajectory model previously studied in an MSHMM, and should be able to overcome the limitation of no continuity constraints between adjacent segments in a piecewise linear trajectory model.

In the HMM-based TTS system built in this research, in which the 12 PFS control parameters and their time derivatives are used as feature vectors in HMMs, two kinds of formant synthesiser control parameter sequences have been investigated:

1. piecewise constant 12 PFS control parameter sequences derived directly from HMMs based on mean vectors and self-transition probabilities,
2. non-linear 12 PFS control parameter sequences, obtained by using the speech parameter generation algorithm based on the same state sequence as in 1.

The Holmes parallel formant synthesiser is employed to synthesise speech from these two types of synthesiser control parameter sequences.

Both subjective and objective tests were conducted to evaluate the synthetic speech in terms of intelligibility, overall quality and perceptual similarity between original speech and synthetic speech generated by using speaker adaptation techniques (both MLLR and MAP are used). Twenty human listeners participated in the subjective tests. Based on the experiment results and statistical significance analysis, the following conclusions are reached:

- Overall, the quality of the synthetic speech is low. The reasons and suggestions for further improvement will be discussed in Section 8.2. The poor overall quality poses a major challenge in virtually all subjective listening experiments, especially in the mimicry test in which test participants are required to recognise speakers.

- For both intelligibility and naturalness tests, despite the low quality of the synthetic speech, the use of non-linear formant synthesiser control parameter trajectories leads to *statistically significant* improvement in performance, relative to the use of piecewise constant PFS control parameter sequences. This demonstrates the benefit of using a non-linear trajectory model of formant dynamics for speech synthesis.

- There is few convincing evidence in the subject mimicry test in support of this particular type of non-linear trajectory model in generating adapted synthetic speech using conventional adaptation techniques such as MLLR and MAP.

The mimicry test is worth mentioning here since it does raise many interesting questions. To reduce the effect of the poor overall quality of the synthetic speech, an 'intermediate' type of synthetic speech, called semi-synthetic speech, is introduced to assist the comparison between original speech and synthetic speech. The semi-synthetic speech is generated by copy synthesis based on the synthesiser control parameters derived directly from the original speech. Even so, the similarity results between original and semi-synthetic speech have shown to be disappointing (on average, 56.8%, when the same sentence is used in both original and semi-synthetic speech). When different sentences are used for original and semi-synthetic speech, the similarity scores drops futher to about 33%. On the other hand, the comparison between semi-synthetic speech and synthetic speech did not show the expected trend that the similarity increases as the number of adaptation utterances increases. It is suspected that the limitation on the maximum number of adaptation utterances (maximum

10 utterances per speaker in TIMIT) and the poor overall quality of the synthetic speech contribute most to the poor result. In addition, these results seem to conform with the recent NIST human-assisted-speaker-recognition evaluation results (Greenberg et al. 2011), which also show that humans are not very good at recognising who is speaking in difficult cases. PESQ results show that model adaptation can improve the quality of the synthetic speech.

8.1.2 Speech recognition

A key contribution of this research is the development of a new class of nonlinear/linear MSHMMs, and the evaluation of the new model through extensive recognition experiments. A nonlinear/linear MSHMM is an extended version of a linear/linear MSHMM, in which speech dynamics are modelled as non-linear trajectories in the formant-based intermediate layer, which are then transformed into the acoustic layer via a set of one or more linear formant-to-acoustic mappings. Compared to a linear/linear MSHMM, the main advantage of a nonlinear/linear MSHMM is the use of an improved, nonlinear trajectory model of formant dynamics, which can overcome the limitation of no continuity constraints between piecewise linear trajectories in a linear/linear MSHMM. Such a non-linear trajectory model of dynamics seeks to provide an active account for the coarticulation and formant transitions, which we believe are important to describe the properties of speech patterns.

The new nonlinear/linear MSHMM is tested against a linear/linear MSHMM based on the N-best list rescoring paradigm on TIMIT. Both context-dependent and context-independent MSHMMs, combined with five different formant-to-acoustic mapping schemes, are used in the rescoring experiments.

The rescoring results show that the use of non-linear formant trajectories in MSHMMs achieves a reduction in phone error rate in most cases, with at most a 4.9% relative reduction in phone error rate (in the case of monophone, mapping 6B, see Table 7.10 on page 166), compared to MSHMMs using linear formant trajectories. Statistical significance tests show that nonlinear/linear MSHMMs give significant improvements in phone accuracy compared to linear/linear MSHMMs when context-independent models are used under mapping scheme 6B and 49E. Moreover, the recognition phone accuracy of nonlinear/linear MSHMMs can be further improved by re-estimation of the formant-to-acoustic mapping using the non-linear formant trajectories generated for the training utterances.

The rescoring results for context-independent and context-dependent MSHMMs follow different patterns. It has been noticed that context-independent MSHMMs benefit more than context-dependent MSHMMs from the use of non-linear formant trajectories in the intermediate layer. To find out the role of nonlinear trajectory model in the MSHMM, comparison of rescoring results is made between monophone and triphone MSHMMs under three conditions: 1) piecewise linear trajectories are used in both monophone and triphone models, 2) nonlinear trajectories are used in monophones and piecewise linear trajectories are retained in triphones, and 3) nonlinear trajectories are used in both monophones and triphones. The main conclusions drawn from the comparison are:

- Under condition 1), context-dependent models outperform context-independent models, with three mapping schemes giving statistically significant improvements. This is expected since context-dependent models contain contextual information that context-independent models do not.

- Under conditions 2) and 3), there are no statistically significant differences between rescoring results of context-independent and context-dependent MSHMMs. In particular, under condition 2), monophone MSHMMs using nonlinear trajectories have been shown to produce similar level of performance than triphone MSHMMs using piecewise linear trajectories. These results demonstrate that the use of non-linear smooth formant trajectories in the intermediate layer of MSHMMs is accounting for contextual effects (e.g., coarticulation) that triphones are designed to accommodate, so that the use of context-dependent models now has less effect.

Experiment results on the TIMIT core test set also demonstrate the superiority of using a non-linear trajectory model for MSHMMs. When the reference transcription are included in the N-best list, a 46% reduction in PER was achieved for non-linear/linear CD MSHMMs relative to linear/linear CD MSHMMs under the mapping scheme 49E. The best phone accuracy (70.4%) is comparable to the conventional CD HMMs system, though is lower than the state-of-the-art.

8.2 Suggestions for future work

In this research, the investigation of using a non-linear trajectory model of speech dynamics has yielded some promising results. There is certainly room for further research in the future.

8.2.1 Generating non-linear trajectories directly from MSHMMS

In this research, all non-linear formant trajectories are generated from an auxiliary set of conventional HMMs that are trained on 12 PFS control parameters and their time derivatives. For a given state sequence, this model set is designed to provide essential model parameters (e.g., means, covariance matrices) to compute the non-linear smooth trajectory based on the speech parameter generation algorithm.

In most cases, however, the state sequence used in calculating the nonlinear trajectory for each hypothesis of the N-best list is determined based on another set of MFCC-based standard HMMs which is used to generate the N-best list. These state sequences are called constrained state sequences due to the use of phone boundary timing in the forced alignment. In linear/linear monophone MSHMMs rescoring experiments, a different type of state sequence, referred to as unconstrained state sequences, is also studied, which is generated based on MSHMMs without using phone boundaries. It has been shown that the use of unconstrained state sequences can result in statistically significant improvement in recognition performance under certain mapping schemes, relative to the use of constrained state sequences.

Based on the above consideration, it is desirable that everything, including state sequences and non-linear trajectories, can be generated directly from a set of MSHMMs. In doing so, MSHMMs can be directly used to generate a sequence of set of nonlinear, smoothed

12 PFS control parameters for speech synthesis. The determination of the state sequence, i.e., the unconstrained state sequence, given an N-best list hypothesis and the MSHMM model set is straightforward. However, the main difficulty is how to apply the speech parameter generation algorithm directly to MSHMMs, in particular the definition of the transform $\boldsymbol{W}$ as in Equation (5.7) (i.e., $\boldsymbol{O} = \boldsymbol{W}\boldsymbol{C}$) on page 86.

In the current implementation of (linear/linear) MSHMMs, only static features (13 MFCCs) are used in the acoustic domain so that it is impossible to define $\boldsymbol{W}$ in the acoustic domain. In the intermediate formant domain, each state is specified by a mean vector and a slope vector, whose dimensions are the same as that of the intermediate space. One possibility is to treat the means as the static features, and slopes as the delta parameters. However, both of the mean and slope vectors are estimated based on the EM algorithm during model training and their relationship can not be simply determined using a deterministic and fixed transform, as defined by Equation (5.7) in the original version of the speech parameter generation algorithm. In fact, each state in a linear/linear MSHMM would have a different transform between the mean and slope vectors. Therefore, a possible research question is how to redefine the relationship between the mean and slope vectors in an MSHMM and how to adapt the speech parameter generation algorithm accordingly to this new relationship?

8.2.2 Improving overall synthetic speech quality

Although maximum overall quality of the synthetic speech is not the objective of this research, there are a number of ways to improve the overall quality of the synthetic speech. It is believed that the improved quality of synthetic speech would yield some more positive results for the synthetic speech evaluation experiments, especially for the mimicry test. The synthetic speech is generated using the HMS parallel formant synthesiser, which can produce high quality synthetic speech given appropriate synthesiser control parameters. Therefore, efforts should be guided to improve the quality of the formant synthesiser control parameters.

The formant synthesis control parameters used in this research are generated from HMMs, and the HMMs are trained on 12 PFS control parameters, which in turn are computed using the Holmes formant analyser. Therefore, there are two possible directions to improve the quality of the synthesiser control parameters. One option is to improve the accuracy of the formant estimation by improving the Holmes formant analyser. Holmes formant analyser was written to run on a very low powered computer, and the design of the formant analysis algorithm is compromised by this. With more computing power, it should be possible to, for example, increase the size of the codebook of pre-labeled spectra and formant frequencies so that more robust estimation of formant frequencies can be achieved. The second possible direction is to improve the quality of the acoustic models. For example, (Zen et al. 2007) found that the use of the trajectory HMM improved the naturalness of synthetic speech, relative to standard HMMs.

Another major limitation of the current speech synthesis system is the lack of a proper prosodic structure. Prosodic structure includes durational structure, the appropriate insertion of pauses, intonation and stress. No synthesis system can produce natural sounding synthetic speech unless an appropriate prosodic structure is incorporated. One possibility to extract these prosodic factors is via feature extraction functions of the Festival speech synthesis system.

8.2.3 Non-linear/non-linear MSHMMs

In the context of a multiple-level segmental HMM, both a simple linear/linear MSHMM (Russell & Jackson 2005, Russell et al. 2007, see Figure 1.1 on page 6) and a more advanced non-linear/linear MSHMMs (Hu & Russell 2008, Hu & Russell 2010, see Figure 7.1 on page 139), have been investigated. The former uses piecewise linear trajectories in the formant-based intermediate layer to model dynamics, while the latter uses non-linear continuous trajectories to account for formant dynamics. In both cases, one or more linear formant-to-acoustic mappings are used to transform the piecewise linear or non-linear formant trajectories in the intermediate layer into the acoustic layer.

The main drawback of the above approaches is the use of linear formant-to-acoustic mapping, which is estimated based on matched acoustic and formant data (see Section 4.4.1).

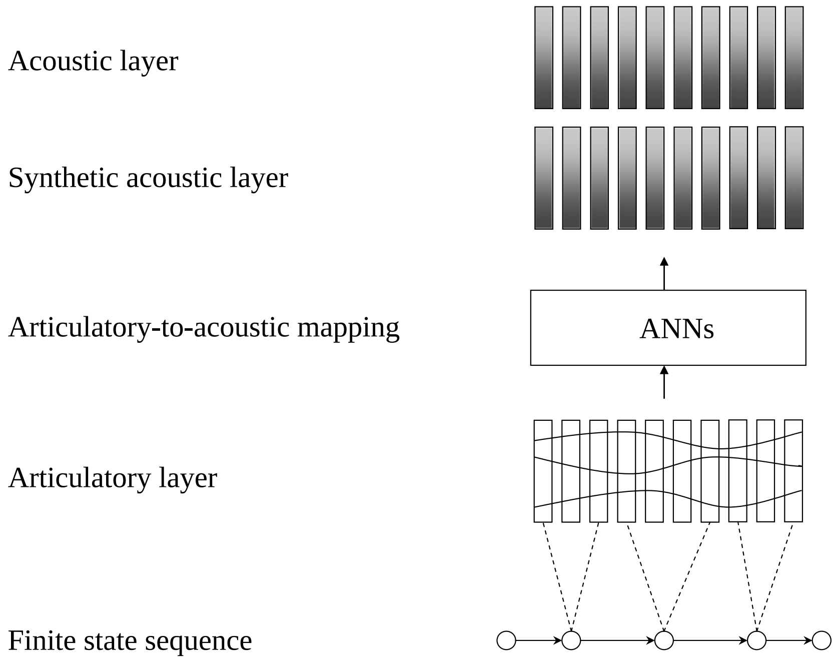


Figure 8.1: A non-linear/non-linear multiple-level segmental HMM in which the dynamics are modelled as non-linear trajectories in the intermediate layer and transformed into the acoustic layer using a non-linear articulatory-to-acoustic mapping.

Several studies have shown that the relationships between formants and spectrum-based acoustic representation are indeed non-linear (Richards & Bridle 1999, Russell et al. 2007). Therefore, a possible direction for future work is to develop a non-linear/non-linear MSHMM, as shown in Figure 8.1, in which formant dynamics are modelled as non-linear trajectories in the intermediate layer and then transformed into the acoustic layer using a non-linear formant-to-acoustic mapping in the form of artificial neural networks.

In view of the unsuitability of a linear formant-to-acoustic mapping, research on alternative, non-linear mapping has been described by a number of authors. Examples include the use of one MLP (Richards & Bridle 1999) or more MLPs (Deng 1998, Deng & Ma 2000, in which 10 MLPs are used) in hidden dynamic models to transform dynamics in the hidden layer to the acoustic layer (see Section 2.4.4 on page 49). Research at the University of Birmingham has investigated the use of one or more RBF networks (Jackson et al. 2002, Lo & Russell 2003, Lo 2004) for the non-linear formant-to-acoustic mapping, which yielded modest improvements.

Training an MLP- or RBF-based non-linear formant-to-acoustic mapping using matched formant and acoustic data is straightforward, though the optimisation of the model trajectory parameters is not. In (Deng 1998), the learning algorithm for the model parameters is based on the generalised EM algorithm with E-step accomplished by extended Kalman filtering (EKF) (a review of the Kalman filtering in speech recognition is provided in (Ostendorf et al. 1996)). It is possible to investigate these techniques in the context of our multiple-level segmental HMMs.

The non-linear/non-linear MSHMMs shoud have theoretical advantages of both the non-linear trajectory model of dynamics and the non-linear formant-to-acoustic mapping. Under such a model, contextual effects and coarticulation could be modelled naturally in the formant-based intermediate layer, and the relationships between the formant and acoustic descriptions of speech are represented more accurately. Provide that the intermediate layer is based on the 12 PFS control parameters, the resulting non-linear/non-linear MSHMMs has the potential to provide both high accuracy speech recognition and high quality speech synthesis.

Obviously, the introduction of the non-linear mappings, combined with the non-linear trajectories, could lead to a tremendous increase in computational load. The N-best list rescoring scheme could be initially used to evaluate these new non-linear/non-linear MSHMMs before an effort is made to develop an appropriate decoder.

8.2.4 Gaussian mixture MSHMMs

The MSHMM presented in this thesis uses a single fixed trajectory in the intermediate layer to model formant dynamics and a single Gaussian PDF in the acoustic layer to model acoustic variability. Modern LVCSR system usually employ Gaussian mixture densities to improve acoustic modelling. Another possible direction for future work is to build a Gaussian mixture MSHMM.

As a starting point, consider a linear PTSHMM, in which a speech segment is characterised by a 'noisy' linear trajectory (see Section 2.3.3). For simplicity, suppose that the trajectories are constant (i.e., slope=0). In this model a state i has two PDFs:

1. The extra-segmental variability PDF, b_i, which is assumed to be Gaussian with mean $\boldsymbol{\nu}_i$ and variance $\boldsymbol{\eta}_i$.

2. The intra-segmental variability PDF, whose mean is the trajectory value and variance is $\boldsymbol{\Sigma}_i$.

Under this model, the joint density of a τ-length speech segment $\boldsymbol{o}_1^\tau = \{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_\tau\}$ and a trajectory $\boldsymbol{f}$ given state i has the form

$$p(\boldsymbol{o}_1^\tau, \boldsymbol{f}) = p(\boldsymbol{o}_1^\tau | \boldsymbol{f}) p(\boldsymbol{f}) = p(\boldsymbol{f}) \prod_{t=1}^{\tau} p(\boldsymbol{o}_t | \boldsymbol{f}(t)), \tag{8.1}$$

and

$$p(\boldsymbol{o}_1^\tau) = \int_{\boldsymbol{f}} p(\boldsymbol{o}_1^\tau, \boldsymbol{f}) d\boldsymbol{f} = \int_{\boldsymbol{f}} p(\boldsymbol{f}) \prod_{t=1}^{\tau} p(\boldsymbol{o}_t | \boldsymbol{f}(t)) d\boldsymbol{f}, \tag{8.2}$$

where the integral is taken for all possible trajectories.

Normally the extra-segmental density $p(\boldsymbol{f})$ is assumed to be Gaussian, i.e., $p(\boldsymbol{f}) = \mathcal{N}(\boldsymbol{f} | \boldsymbol{\nu}_i, \boldsymbol{\eta}_i)$. It is straightforward to extend this single-mixture model to a GMM. Letting

$$p(\boldsymbol{f}) = \sum_{k=1}^{K} w_{ik} \mathcal{N}(\boldsymbol{f} | \boldsymbol{\nu}_{ik}, \boldsymbol{\eta}_{ik}), \tag{8.3}$$

where w_{ik}, $\boldsymbol{\nu}_{ik}$ and $\boldsymbol{\eta}_{ik}$ are mixture weight, mean and variance for the i-th mixture component, respectively, and $\sum_{k=1}^{K} w_{ik} = 1$. Then substituting this mixture PDF in Equation (8.2)

$$p(\boldsymbol{o}_1^\tau) = \sum_{k=1}^{K} w_{ik} \int_{\boldsymbol{f}} \mathcal{N}(\boldsymbol{f} | \boldsymbol{\nu}_{ik}, \boldsymbol{\eta}_{ik}) \prod_{t=1}^{\tau} p(\boldsymbol{o}_t | \boldsymbol{f}(t)) d\boldsymbol{f}. \tag{8.4}$$

In this case, instead of modelling a segment as a noisy trajectory, a segment is modelled as a set of K alternative noisy trajectories, each with its own prior probability (i.e., the GMM weights w_{ik}).

Modelling the intra-segmental variability PDF using a GMM is more difficult. In the standard model this PDF is centred on the current trajectory value $\boldsymbol{f}(t)$. For a single Gaussian PDF this is simple to achieve by just choosing the current trajectory value $\boldsymbol{f}(t)$ to be the PDF mean (as can be seen in Equation 8.1). However, a GMM does not have a single mean so this is not possible. An alternative view of the standard model is that the intra-segmental PDF is just a 'noise' PDF (with mean zero) and we add the current trajectory value to the mean at each point of the trajectory. So it may be possible to define the 'noise' process to be a fixed GMM, and then at each time t add the trajectory value $\boldsymbol{f}(t)$ to each of the GMM means (Russell 2012).

In a fixed trajectory SHMM (see Section 2.3.4), the extra-segmental PDF variance is fixed at zero so that the trajectory associated with a state is fixed. For a fixed trajectory model, Equation (8.2) becomes

$$p(\boldsymbol{o}_1^\tau) = p(\boldsymbol{o}_1^\tau | \boldsymbol{f}) = \prod_{t=1}^{\tau} p(\boldsymbol{o}_t | \boldsymbol{f}(t)), \tag{8.5}$$

Where $\boldsymbol{f}$ is the fixed trajectory associated with the state. For the GMM equivalent of a fixed trajectory SHMM, each state would be associated with a set of K trajectories $\{\boldsymbol{f}_1, \dots, \boldsymbol{f}_K\}$ each with a prior probability $\{w_{i1}, \dots, w_{iK}\}$ and $\sum_{k=1}^{K} w_{ik} = 1$, such that

$$p(\boldsymbol{o}_1^\tau) = \sum_{k=1}^{K} w_{ik} p(\boldsymbol{o}_1^\tau | \boldsymbol{f}_k) = \sum_{k=1}^{K} w_{ik} \prod_{t=1}^{\tau} p(\boldsymbol{o}_t | \boldsymbol{f}_k(t)). \tag{8.6}$$

Now assume that the trajectory $\boldsymbol{f}_k$ is realised in the formant-based intermediate layer and projected to the acoustic layer using a formant-to-acoustic mapping $\boldsymbol{W}$, then the above equation can be written as

$$p(\boldsymbol{o}_1^\tau) = \sum_{k=1}^{K} w_{ik} \prod_{t=1}^{\tau} p\Big(\boldsymbol{o}_t | \boldsymbol{W}\big(\boldsymbol{f}_k(t)\big)\Big) = \sum_{k=1}^{K} w_{ik} \prod_{t=1}^{\tau} \mathcal{N}\Big(\boldsymbol{o}_t | \boldsymbol{W}\big(\boldsymbol{f}_k(t)\big), \Sigma_i\Big), \tag{8.7}$$

where Σ_i is the covariance matrix for state i. Equation 8.7 defines a Gaussian mixture MSHMM which could be investigated further in the future. It is assumed that the principle of the model parameter estimation algorithm for the existing MSHMMs still applies.

As with conventional HMMs, one would expect the above mixture model to give superior performance, relative to the single mixture model. However, the number of parameters of the mixture model given by Equation 8.7 goes up proportionally to the number of mixture components K, and hence more training data is required to accurately estimate model parameters, especially when context-dependent models are used. The underlying motivation for probabilistic trajectory-based SHMMs is to replace the reliance on statistical models with large numbers of parameters by a more compact model that aims to capture speech dynamics more accurately. By following the above GMM-based routine we are returning to trying to solve the problem 'conventionally' by using large numbers of parameters.

8.3 Concluding remark

Some technologies portrayed as common in *2001: A Space Odyssey* have indeed materialised in the 2000s, for example, small, portable, flat-screen devices like the iPad. However, the challenge of building an intelligent machine that can speak naturally and respond properly to spoken language, as displayed by HAL, remains enormous. Of course, there are many possible directions one can choose to attack the problem. This study made an attempt to develop a unified model for both speech recognition and synthesis. The search for the ultimate speech models continues...

Appendix A

TIMIT phone set

A.1 TIMIT 49-phone set

Phone	Example	Folded	Phone	Example	Folded
iy	*beat*		en	*button*	
ih	*bit*		ng	*sing*	eng
eh	*bet*		ch	*church*	
ae	*bat*		jh	*judge*	
ix	*roses*		dh	*they*	
ax	*the*		b	*bob*	
ah	*butt*		d	*dad*	
uw	*boot*	ux	dx	*butter*	
uh	*book*		g	*gag*	
ao	*about*		p	*pop*	
aa	*cot*		t	*tot*	
ey	*bait*		k	*kick*	
ay	*bite*		z	*zoo*	
oy	*boy*		zh	*measure*	
aw	*bough*		v	*very*	
ow	*boat*		f	*fief*	
l	*led*		th	*thief*	
el	*bottle*		s	*sis*	
r	*red*		sh	*shoe*	
y	*yet*		hh	*hay*	hv
w	*wet*		cl(sil)	(unvoiced closure)	pcl,tcl,kcl,qcl
er	*bird*	axr	vcl(sil)	(voiced closure)	bcl,dcl,gcl
m	*mom*	em	epi(sil)	(epinthetic closure)	
n	*non*	nx	sil	(silence)	h#,#h,pau
			q		

A.2 TIMIT allowable confusions

The table below shows the TIMIT allowable confusions (Lee & Hon 1989) when evaluating the recognition results, leading to the TIMIT 39-phone set.

TIMIT equivalent labels for evaluation
{sil, cl, vcl, epi}
{el, l}
{en, n}
{sh, zh}
{ao, aa}
{ih, ix}
{ah, ax}

Appendix B

The DRT 96 word-pairs list

The tables below contais 96 word-pairs used in the DRT test, which are taken from (Pratt 1984).

Voicing		Nasality		Sustension	
Voiced	**Unvoiced**	**Nasal**	**Oral**	**Sustained**	**Interrupted**
veal	feel	meat	beat	vee	bee
bean	pean	need	deed	sheet	cheat
gin	chin	mitt	bit	vill	bill
dint	tint	nip	dip	thick	tick
zoo	sue	moot	boot	foo	pooh
dune	tune	news	dues	shoes	choose
voal	foal	moan	bone	those	doze
goat	coat	note	dote	though	dough
zed	said	mend	bend	then	den
dense	tense	neck	deck	fence	pence
vast	fast	mad	bad	than	dan
gaff	calf	nab	dab	shad	chad
vault	fault	moss	boss	thong	tong
daunt	taunt	gnaw	daw	shaw	chaw
jock	chock	mom	bomb	von	bon
bond	pond	knock	dock	vox	box

Sibilation		Graveness		Compactness	
sibilated	**Unsibilated**	**Grave**	**Acute**	**Compact**	**Diffuse**
zee	thee	weed	reed	yield	wield
cheep	keep	peak	teak	key	tea
jilt	gilt	bid	did	hit	fit
sing	thing	fin	thin	gill	dill
juice	goose	moon	noon	coop	poop
chew	coo	pool	tool	you	rue
joe	go	bowl	dole	ghost	boast
sole	thole	fore	thor	show	so
jest	guest	met	net	keg	peg
chair	care	pent	tent	yen	wren
jab	gab	bank	dank	gat	bat
sank	thank	fad	thad	shad	sag
jaws	gauze	fought	thought	yawl	wall
saw	thaw	bong	dong	caught	taught
jot	got	wad	rod	hop	fop
chop	cop	pot	tot	got	dot

Appendix C

Publications

Two conference papers were published during the period of this thesis work:

- Hu, H. & Russell, M.J. (2008), Speech recognition using non-linear trajectories in a formant-based articulatory layer of a multiple-level segmental HMM, *in* 'Proc. INTERSPEECH 2008', Brisbane, Australia, pp. 2422-2425.
- Hu, H. & Russell, M.J. (2010), Improved modelling of speech dynamics using non-linear formant trajectories for HMM-based speech synthesis, *in* 'Proc. INTERSPEECH 2010', Makuhari, Chiba, Japan, pp. 821-824.

These two papers are also attached for reference.

Speech recognition using non-linear trajectories in a formant-based articulatory layer of a multiple-level segmental HMM

Hongwei Hu, Martin J. Russell

Department of Electronic, Electrical and Computer Engineering,
University of Birmingham, Birmingham B15 2TT, UK
hwh400@bham.ac.uk, m.j.russell@bham.ac.uk

Abstract

This paper describes how non-linear formant trajectories, based on ‘trajectory HMM’ proposed by Tokuda *et al.*, can be exploited under the framework of multiple-level segmental HMMs. In the resultant model, named a *non-linear/linear* multiple-level segmental HMM, speech dynamics are modeled as non-linear smooth trajectories in the formant-based intermediate layer. These formant trajectories are mapped into the acoustic layer using a set of one or more linear mappings. The N-best rescoring paradigm is employed to evaluate the performance of the non-linear formant trajectories. The rescoring results on TIMIT corpus show that the introduction of non-linear formant trajectories results in improvement on recognition phone accuracy compared with linear trajectories.

Index Terms: speech recognition, non-linear formant trajectories, segmental HMMs

1. Introduction

The purpose of this paper is to determine whether the use of smooth, non-linear formant trajectories in a multiple-level segmental HMM (MSHMM) can result in improved speech recognition performance compared with a MSHMM using linear formant trajectories. In a multiple-level *linear/linear* segmental HMM previously presented in [1], the relationship between the underlying symbolic and surface acoustic (e.g. MFCCs) representations of a speech signal is regulated by an intermediate ‘articulatory’ layer. Figure 1 (a) shows the structure of a linear/linear MSHMM. States of the underlying Markov process are associated with piecewise linear trajectories in the intermediate layer, which are mapped into the acoustic layer using a linear articulatory-to-acoustic mapping. The results of phonetic classification experiments on TIMIT show that, even with this simple linear/linear MSHMM system, speech recognition performance can achieve the upper bound of an appropriate fixed linear-trajectory acoustic segmental HMM (FT-SHMM), which, in turn, can outperform a conventional HMM [2]. It was hoped that further improvements in performance relative to a conventional HMM could be achieved by using appropriate non-linear models of dynamics, alternative articulatory representations or non-linear articulatory-to-acoustic mappings.

One of the limitations of the linear/linear MSHMM is the use of linear trajectories. A linear trajectory is simply characterized by a mid-point and a slope vector, which are of the same dimension of the intermediate layer. No continuity constraints are applied across the segment boundaries. Although a piecewise linear model provides an adequate ‘passive’ approximation to the formant trajectories, it does not capture the active dynamics of the articulatory system. Many alternative intermediate-layer models of dynamics have been proposed, e.g., [3, 4, 5, 6, 7, 8]. In addition, it has been noted that a linear articulatory-to-acoustic mapping is inadequate in general [3].

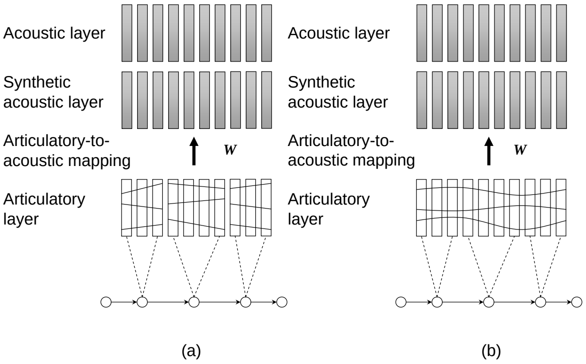


Figure 1: *A multiple-level segmental HMM (MSHMM) in which the relationship between the symbolic and acoustic representations of a speech signal is regulated by an intermediate formant-based layer: (a) a linear/linear MSHMM, (b) a non-linear/linear MSHMM*

In this paper, non-linear formant trajectories are generated based on Tokuda’s ‘trajectory HMM’ method [9]. For consistency with the linear/linear MSHMM described in [1], the articulatory-to-acoustic mappings are also linear, but the mapping parameters are re-estimated based on the non-linear trajectories data. The resultant model is referred to as a *non-linear/linear* MSHMM, and is shown in Figure 1(b).

Five ‘formant-to-acoustic’ mapping schemes, based on different phone categories [1, 10], are considered in order to investigate their effects on the system performance. Phonetic recognition experiments on TIMIT are performed to compare the performance of non-linear trajectories and linear trajectories. As an appropriate decoder for non-linear/linear MSHMMs is not yet available, an N-best list rescoring paradigm, where $N = 1,000$, is employed to evaluate the effectiveness of the non-linear formant trajectories.

The rest of this paper is organized as follows. Section 2 describes the formant-based intermediate layer. Section 3 briefly explains the speech parameter generation algorithm to produce non-linear trajectories. Speech recognition experiments and results are shown in Section 4 and 5. Conclusions and future work are presented in the final section.

2. Formant-based intermediate layer

The intermediate layer presented in [1] is based on formant frequencies, ranging from the simplest form which just consists of the first three formant frequencies to a 12 dimensional representation using the 12 parallel formant synthesiser (PFS) control parameters. Experimental results show that the performance improves either as the dimension of the intermediate representation or the number of mappings is increased [10]. This paper concentrates on the 12 PFS control parameter representation, referred to as *12PFS*, which are produced by Holmes-Mattingley-Shearme (HMS) formant analyser [11]. The 12 PFS control parameters are listed in Table 1.

No.	Parameter description
1	FN: Freq. of 'low freq.' formant (default 250Hz)
2	ALF: Amplitude of FN in dB
3	F1: Freq. of first formant (in 25 Hz steps)
4	A1: Amplitude of first formant in dB
5	F2: Freq. of second formant (in 50 Hz steps)
6	A2: Amplitude of second formant in dB
7	F3: Freq. of third formant (in 50 Hz steps)
8	A3: Amplitude of third formant in dB
9	AHF: Amp. in high freq. region in dB
10	V: Degree of voicing
11	F0: Fundamental freq. on logarithmic scale
12	MS: Glottal-pulse mark-space ratio

Table 1: *The 12 parallel formant synthesiser (PFS) control parameter representation, 12PFS, in the intermediate layer of a MSHMM.*

It has to be confessed that this is a relatively simple 'articulatory' representation compared to those presented in [3, 4, 5, 6, 7, 8]. Also, strictly speaking, it is only implicitly articulatory by nature. However, a major motive of incorporating such an intermediate layer is to allow speech dynamics to be modelled simply and directly in an articulatory-related space, which is capable of supporting both recognition and synthesis. Indeed, it has been demonstrated that given a sequence of appropriate PFS control parameters, extremely high quality speech can be produced using a parallel formant synthesiser. This suggests that a MSHMM, whose intermediate layer is the *12PFS* representation, can be used for trainable model-based speech synthesis. This motivates the concept of 'unified' model for speech recognition and synthesis proposed in [12].

3. Trajectory modelling

The non-linear formant trajectories in this paper are generated based on the 'trajectory HMM' method [9]. Most state-of-the-art automatic speech recognition (ASR) systems use static and dynamic features (e.g. delta and delta-delta coefficients) to accommodate temporal dynamics. Although the employment of dynamic features improves the performance of HMM-based speech recognizers, their use leads to inconsistencies in a conventional HMM, where it is assumed that both the static and dynamic parameters can be simultaneously constant and non-zero. The motivation for 'trajectory HMM' is to alleviate this inconsistency between static features and dynamic features of a conventional HMM. A trajectory, which is most consistent with the static and dynamic constraints is synthesised in the static feature vector space. In fact, the new trajectory is exactly the same as the speech parameter trajectory generated by using the speech parameter generation technique in [13]. A detailed description of the algorithm appears in [13], where it is referred to as case 1, but a brief explanation is given here for completeness.

Let $O = \{o_1, ..., o_T\}$ be a sequence of speech observations and $S = \{s_1, ..., s_T\}$ a fixed state sequence of an HMM M. Assume that the speech vector o_t consists of both a static feature vector c_t and dynamic feature vectors Δc_t, $\Delta^2 c_t$. The delta and delta-delta coefficients are computed using the following formulae, where θ is set to 1 in this paper.

$$\Delta c_t = \frac{c_{t+\theta} - c_{t-\theta}}{2\theta} \tag{1}$$

$$\Delta^2 c_t = \frac{\Delta c_{t+\theta} - \Delta c_{t-\theta}}{2\theta} \tag{2}$$

Let W be the linear transform matrix which transforms the sequence of static parameter vectors C into an 'augmented' sequence of static plus dynamic vectors O. Then the above formulae can be written as

$$O = WC \tag{3}$$

For a given state sequence S, the speech parameter generation problem is to determine the parameter sequence C which maximize $P(O|S, M)$ with respect to C under the constraints (3). By setting

$$\frac{\partial log P(WC|S, M)}{\partial C} = 0 \tag{4}$$

a set of equations are obtained. For detailed representation of the transform matrix W and equation (4), refer to [13]. The non-linear trajectories used in this paper are obtained by solving equation (4). This speech parameter generation technique forms the basis of HMM-based speech synthesis described in [13].

4. Experiment methods

4.1. Speech data

The male part of the TIMIT corpus is used for all experiments. The training data includes all male utterances from the TIMIT training set (3,252 utterances) and the test data includes utterances from male speakers in TIMIT core test set (128 utterances, excluding 'sa1' and 'sa2' sentences). 13 MFCCs, including zeroth, are used as the acoustic features for MSHMMs training and evaluation. They are produced using HTK with 25 ms window, 10ms frame rate. 39 MFCCs, including Δ, Δ^2 coefficients are also used in the experiments. The 12 PFS control parameters are produced by the HMS formant analyzer, converted to HTK format and augmented with Δ and Δ^2 coefficients, resulting in a representation of 36 dimensional features.

4.2. N-best list

An N-best list where $N = 1,000$ is generated using a set of standard decision-tree based triphone HMMs built using HTK, which consists of 1,000 most likely hypotheses for each test utterance. The triphone model set is built on 39 MFCCs (including Δ and Δ^2 coefficients). The baseline phone accuracy (1 best decoding) is 65.7%. The upper bound phone accuracy for the N-best list is 77.9%, which is based on the most accurate match from the 1,000 alternatives with the reference label files. Therefore, this is the theoretical maximum score the following N-best rescoring experiments can reach. The optimal state sequence for each N-best list hypothesis can be produced using forced alignment with HTK.

4.3. Model sets

Five linear/linear monophone MSHMM model sets are built on the TIMIT corpus using 13 MFCCs (static features only), based on different articulatory-to-acoustic mapping schemes. The MSHMM model parameters are optimized using an estimation-maximization (EM) scheme based on segmental Viterbi decoding [1], implemented as part of the 'SEGVit' toolkit.

A	all data

B	vowels, {hh,l,r,w,y}, nasals, {dh,f,s,sh,th,v,z,zh}
	{ch,jh}, {b,cl,d,vcl,dx,epi,g,k,p,q,sil,t}

C	vowels, {epi,q,sil}, {hh,l,r,w,y}, nasals, {ch,s}
	{sh,f,th}, {dh,v,zh}, {jh,z}, {cl,k,p,t}, {b,d,vcl,dx,g}

D	vowels, {epi,q,sil}, {dx,el,l,r,w,y}, nasals, {vcl}, {cl}
	{b,d,g,jh}, {ch,k,p,q,t}, {dh,v,z,zh}, {f,hh,s,sh,th}

E	49 individual phones

Table 2: *Definitions of the phone categories: B. linguistic categories; C. as in [6]; D. discrete articulatory regions. 'nasals' and 'vowels' denote the sets {en,m,n,ng}, and {aa,ae,ah,ao,aw,ax,ay,eh,el,er,ey,ih,ix,iy,ow,oy,uh,uw} respectively.*

The formant-to-acoustic mapping schemes considered in this paper include all five mapping schemes described in [1], ranging from a single 'phone-independent' mapping to 49 'phone-dependent' mappings. The five mapping schemes are referred to as 1A, 6B, 10C, 10D and 49E respectively (number of mappings followed by phone partition), as is shown in Table 2. The mappings are estimated by minimizing the error between matched sequences of formant and acoustic data. For example, assume that $f = \{f_1, ..., f_T\}$ and $y = \{y_1, ..., y_T\}$ are formant and acoustic sequences corresponding to a particular phone class, then the mapping W is calculated by minimizing the error E as follows:

$$E = \sum_{t=1}^{T} \|Wf_t - y_t\|^2 \qquad (5)$$

In practice, the mappings are pre-computed using the TIMIT corpus before estimation of the MSHMM model parameters. A bias, which is set to '1', is added to 12 PFS control parameters to accommodate offsets.

4.4. Segment probability calculation

Suppose that the unknown utterance corresponds to a sequence of acoustic feature vectors $y = y_1, ..., y_T$. For a given entry in the N-best list, this utterance corresponds to a state sequence $s = \tau_1 \times s_1, \tau_2 \times s_2, ..., \tau_N \times s_N$, where $\tau_1 + \tau_2 + ... + \tau_N = T$ and $\tau_n \times s_n$ denotes τ_n repetitions of state s_n. A trajectory f of length T is defined in the intermediate space as $f = f_1, f_2, ..., f_N$, where f_n is a trajectory 'fragment' of length τ_n. In the case of a linear/linear MSHMM each f_n is a linear trajectory, defined by a slope and mid-point value for state s_n, as in [1]. For a non-linear/linear MSHMM f is a single, continuous trajectory obtained using the 'trajectory HMM' method, and f_n is simply the section of f which corresponds to state s_n. The probability of y is then given by:

$$p(y) = \prod_{n=1}^{N} d_n(\tau_n)\{\prod_{t=1}^{\tau_n} N(y_{\phi_n+t}; W_{s_n}(f_n(y_{\phi_n+t})), \sigma_n)\} \qquad (6)$$

where $\phi_n = \sum_{i=1}^{n-1} \tau_i$, $N(y; \mu, \sigma)$ denotes a multivariate Gaussian probability density function (PDF), with mean μ and diagonal covariance σ, and σ_n and d_n are the (acoustic) variance and duration PDF associated with state n. The state duration PDF was uniform with the maximum state duration set to 15 frames ($\tau_{max} = 15$).

4.5. Rescoring with linear/linear MSHMMs

For comparison purpose, the baseline experiment is to rescore the N-best list using linear/linear MSHMMs. Rescoring with linear/linear MSHMMs is quite straightforward. For a given state sequence, equation (6) is used to calculate the probability of a sequence of acoustic features. Each hypothesis in the N-best list is rescored and the one with the highest probability is recorded for evaluation.

It is worth noting here that this paper considers two types of state sequence $s_1, ..., s_N$. One is generated using conventional HMMs, with HTK. Phone boundaries are specified as a consequence of the forced alignment. This is referred to as a 'constrained' state sequence. In this case the MSHMM is forced to use the same state sequence. The other, 'unconstrained' state sequence is produced, ignoring the conventional HMM boundary timing, with 'SEGVit'. Both are used in turn to rescore the N-best list with linear/linear MSHMM. However, only the constrained state sequence is used in the non-linear rescoring experiment.

4.6. Rescoring with non-linear/linear MSHMMs

As the non-linear trajectories are realized in the articulatory-based space, a set of conventional monophone HMMs (single Gaussian) are built on 36 dimensional PFS control parameters (12 PFS control parameters plus Δ and Δ^2 coefficients) using HTK. To rescore with non-linear/linear MSHMMs, non-linear trajectories are generated for each N-best list hypothesis. The state sequence is a constrained state sequence, which is the same as that used in the constrained linear/linear MSHMMs rescoring. Given this state sequence, the non-linear trajectory can be directly generated using the technique described in Section 3.

5. Experiment Results

5.1. Linear/linear MSHMMs rescoring results

Tables 3 and 4 show the rescoring results for linear/linear MSHMMs, where the state sequence used in Table 3 is constrained and the unconstrained state sequence is used in Table 4. The third rows of the tables show the rescoring results after the (correct) reference transcriptions are added to the N-best lists.

Mapping	1A	6B	10C	10D	49E
Phone accuracy	65.7	65.0	66.0	66.3	65.4
Phone acc.(N+1)	68.4	68.5	68.0	70.1	67.3

Table 3: *Phone accuracy of rescoring with linear/linear MSHMMs using constrained state sequence on different mapping schemes*

Table 3 shows that the MSHMM based on mapping '10D' gives the highest phone accuracy, either without or with the reference transcriptions. However, for the unconstrained system Table 4 shows the highest phone accuracy is reached using '49E' scheme. In case reference transcriptions are not included,

Mapping	1A	6B	10C	10D	49E
Phone accuracy	65.3	65.8	66.1	66.4	66.4
Phone acc.(N+1)	67.8	68.4	69.7	68.6	70.4

Table 4: *Phone accuracy of rescoring with linear/linear MSHMMs using unconstrained state sequence on different mapping schemes*

the unconstrained rescoring results (Table 4) are generally better than the corresponding constrained results. Furthermore, the unconstrained rescoring results in Table 4 also show the trend that phone accuracy increases as the number of articulatory-to-acoustic mappings increases.

5.2. Non-linear/linear MSHMMs rescoring results

Mapping	1A	6B	10C	10D	49E
MSHMM(old)	66.1	66.4	66.4	66.4	66.7
Mapping reest.	66.2	66.7	66.7	66.9	67.0

Table 5: *Phone accuracy of rescoring with non-linear/linear MSHMMs using constrained state sequence on different mapping schemes. Mapping reest. means MSHMM mappings are re-estimated.*

Table 5 shows the results of rescoring the N-best list using non-linear/linear MSHMMs. MSHMM(old) means the rescoring is based on the original MSHMM models, while ‘mapping reest.’ indicates that an updated MSHMM model set, whose articulatory-to-acoustic mapping parameters are re-estimated relative to the non-linear trajectories, is used for rescoring.

The mapping re-estimation is also based on equation 5. The original mappings are trained based on the matched sequence of 12 PFS control parameters and 13 MFCCs in the TIMIT training set. However, instead of using the original 12 PFS control parameters, the mappings are re-trained based on a sequence of smooth non-linear trajectories data derived from using the same algorithm discussed in Section 3. For each utterance in the TIMIT training set, non-linear trajectories are generated in the same way with a N-best list hypothesis. Therefore, the re-trained mapping is based on matched sequences of non-linear smooth trajectories in 12PFS space and 13 dimensional MFCC vectors. As is shown in Table 5, the phone accuracies increase for all mapping schemes after the mappings are re-trained. This suggests that the re-estimated mapping provides a better fit between the acoustic features and non-linear formant trajectories.

Because the non-linear/linear rescoring is based on the same constrained state sequence as in Table 3, the comparison should be made between Table 3 and Table 5. It can be seen that non-linear/linear MSHMMs outperform the linear/linear MSHMM in every mapping scheme, with at most a 2% reduction of error rate for mapping ‘6B’. The non-linear/linear MSHMM also outperforms the linear/linear MSHMM whose rescoring is based on unconstrained state sequences (Table 4). Table 5 also shows the trend that performance increases as the number of mappings increases. As can be seen, the highest phone accuracy is 67.0% under the mapping scheme 49E.

6. Conclusions and future work

This paper has shown that the use of non-linear formant trajectories achieves a reduction in phone error rate on TIMIT, although the improvement is modest. Hence it can be argued that the use of non-linear formant trajectories improves the recognition performance, compared with piecewise linear trajectories. This paper also compares the performance of different articulatory-to-acoustic mapping schemes and the constrained and unconstrained state sequences in the rescoring experiments.

The model sets used in this paper comprise monophone MSHMMs. Of course, the results of a triphone experiment would be more interesting and this experiment is ongoing. It is hoped that rescoring the same N-best list using triphone MSHMMs would give a higher score of phone accuracy. However, the monophone results presented in this paper show that the MSHMM performance can be improved by using richer class of non-linear smooth trajectories and more sophisticated acticulatory-to-acoustic mappings.

In the future, a short term goal is to evaluate the non-linear formant trajectories using context-dependent triphone MSHMMs. The N-best rescoring paradigm will still be used at this stage. A longer-term goal is to develop a Viterbi type decoder for the training and evaluation of non-linear/linear MSHMMs.

7. References

[1] Russell, M.J. and Jackson, P.J.B., “A multiple-level linear/linear segmental HMM with a formant-based intermediate layer”, *Computer Speech and Language*, vol. 19, pp. 205–225, 2005.

[2] Holmes, W.J. and Russell, M.J., “Probabilistic-trajectory segmental HMMs”, *Computer Speech and Language*, vol. 13, pp. 3–37, 1999.

[3] Richards, H.B. and Bridle, J.S., “The HDM: a segmental Hidden Dynamic Model of coarticulation”, *Proc. IEEE-ICASSP*, Phoenix, AZ, pp357-360, 1999.

[4] Deng, L. and Braam, D., “Context-dependent Markov model structured by locus equations: applications to phonetic classification”, *J. Acoust. Soc. Am.*, 96(4), pp. 2008–2025, 1996.

[5] Deng, L., “A dynamic, feature-based approach to the interface between phonology and phonetics for speech modelling and recognition”, *Speech communication*, 24(4), pp. 288–323, 1998.

[6] Deng, L. and Ma, J., “Spontaneous speech recognition using a statistical coarticulatory model for the vocal-tract-resonance dynamics”, *J. Acoust. Soc. Am.*, 108(6), pp. 3036–3048, 2000.

[7] Gao, Y., Bakis, R., Huang, J. and Zhang, B., “Multistage coarticulation model combining articulatory, formant and cepstral features”, *International Conference on Spoken Language Proc.*, Beijing, vol. 1, pp. 25–28, 2000.

[8] Zhou, J., Seide, F. and Deng, L., “Coarticulation modelling by embedding a target-directed hidden trajectory model into HMM - modelling and training”, *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Proc.*, Hong Kong, 1, pp. 744–747, 2003.

[9] Tokuda, K., Zen H. and Kitamura T., “Trajectory modeling based on HMMs with the explicit relationship between static and dynamic features”, *Proc. EUROSPEECH 2003*, Geneva, Switzerland, pp. 865–868, 2003.

[10] Russell, M.J., Zheng, X. and Jackson, P.J.B., “Modelling speech signals using formant frequencies as an intermediate representation”, *IET Signal Process*, 1, pp. 43–50, 2007.

[11] Holmes, J.N., Mattingly, I.G. and Shearme, J.N., “Speech synthesis by rule”, *Language & Speech*, 7, pp. 127–143, 1996.

[12] Russell, M.J., “A unified model for speech recognition and synthesis”, University of Birmingham, Edgbaston, Birmingham, UK, 2004.

[13] Tokuda, K., Yoshimura, T., Masuko T., Kobayashi, T. and Kitamura T., “Speech parameter generation algorithms for HMM-based speech synthesis”, *Proc. ICASSP*, vol.3, pp. 1315–1318, 2000.

Improved modelling of speech dynamics using non-linear formant trajectories for HMM-based speech synthesis

Hongwei Hu, Martin J. Russell

School of Electronic, Electrical & Computer Engineering
University of Birmingham, Birmingham, B15 2TT, UK
hwh400@bham.ac.uk, m.j.russell@bham.ac.uk

Abstract

This paper describes the use of non-linear formant trajectories to model speech dynamics. The performance of the non-linear formant dynamics model is evaluated using HMM-based speech synthesis experiments, in which the 12 dimensional parallel formant synthesiser control parameters and their time derivatives are used as the feature vectors in the HMM. Two types of formant synthesiser control parameters, named piecewise constant and smooth trajectory parameters, are used to drive the classic parallel formant synthesiser. The quality of the synthetic speech is assessed using three kinds of subjective tests. This paper shows that the non-linear formant dynamics model can improve the performance of HMM-based speech synthesis.

Index Terms: Dynamics, HMM synthesis, speaker adaptation

1. Introduction

The purpose of this study is to investigate the use of non-linear formant trajectories to improve the dynamics model for speech processing. From this perspective, the starting point of this research is a *linear/linear* multiple-level segmental HMM (MSHMM) [1], in which dynamics are modelled as piecewise constant, *linear* trajectories in a formant-based articulatory layer, as is shown in Figure 1. These linear formant trajectories are then transformed into the acoustic space using one or more *linear* articulatory-to-acoustic mappings. The intermediate layer presented in [1] is based on formant frequencies, ranging from the simplest form which just consists of the first three formant frequencies to a comprehensive representation using the 12 parallel formant synthesiser (PFS) control parameters [2]. Alternative intermediate-layer models of dynamics have also been studied in the past, such as [3, 4], which provide much complex models of dynamics. A major motivation for incorporating such a formant-based intermediate layer into a MSHMM is to allow speech dynamics to be modelled simply and directly in an articulatory related space without compromising the recognition performance. Indeed, both phonetic classification [1] and recognition [5] results on TIMIT show that, even with this simple linear/linear MSHMM system, speech recognition performance can achieve the upper bound of a fixed linear-trajectory acoustic segmental HMM. In other words, the incorporation of such an intermediate layer does not 'hurt'.

One of the limitations of the above linear/linear MSHMM system is the use of linear trajectories to model dynamics. Although a piecewise linear model provides an adequate 'passive' approximation to the formant trajectories, it does not capture the active dynamics of the articulatory system. Moreover, no continuity constraints are considered across the segment boundaries. This motivates the work in [6], which demonstrates that the phone recognition accuracy can be further improved by using non-linear formant trajectories to model dynamics. In [6], the non-linear trajectories are generated by using the speech parameter generation algorithm described in [7].

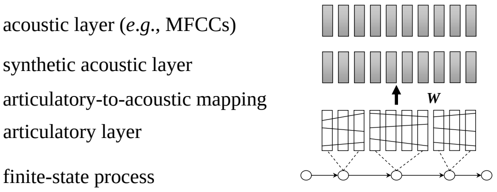


Figure 1: *A linear/linear multiple-level segmental HMM.*

This research builds on the work in [6] to evaluate the performance of the non-linear model of dynamics in a HMM-based speech synthesis paradigm. A simple HMM-based speech synthesis system is built, in which the HMMs are trained on 12 PFS control parameters and their time derivatives. It was hoped that the 12 PFS control parameters, which are conventionally used for formant synthesis, would also be useful for HMM-based speech synthesis, and the non-linear dynamics model presented in [6] for speech recognition would be beneficial for HMM synthesis as well. Both HMM-based speech synthesis [7] and the use of formant information for HMM-based speech synthesis [8] have been studied in the past. In either case, the speech parameter generation algorithm is used to produce smoothed cepstral or formant trajectories. The emphasis of this research, however, is to study the effect of the non-linear dynamics model for HMM-based synthesis, rather than to develop a state-of-the-art synthesis system. Compared with other HMM-based synthesis systems, a novelty of the HMM-based synthesis system developed in this research is the use of the 12 PFS control parameters, which combine both source and filter parameters in a compact form, to train the HMMs and the use of a parallel formant synthesiser to synthesize speech.

This research has implications for developing unified, trainable models which can support both recognition and synthesis within the framework of a MSHMM, whose intermediate layer is based on 12 PFS control parameters. In principle, these parameters are sufficient to create a 'talker table' to define a 'voice' for a formant synthesiser. If speaker adaptation techniques such as MLLR or MAP are used to operate in the formant domain to adapt the model to an individual's speech, the resultant synthetic speech should sound like that individual. Moreover, adaptation in the articulatory layer should result in more interpretable changes in formant frequencies.

The rest of this paper is organized as follows. Section 2 briefly reviews the speech parameter generation algorithm to generate non-linear trajectories. Speech synthesis experiments and results are shown in Section 3 and 4. Conclusions and future work are presented in the final section.

2. Speech parameter generation algorithm

The non-linear formant trajectories used to model dynamics in this research are generated based on the speech parameter generation technique [7]. A detailed description of this algorithm appears in [7], where it is referred to as case 1, and a brief review is provided here for completeness.

Let $\boldsymbol{O} = \{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_T\}$ be a sequence of speech observations and $S = \{s_1, ..., s_T\}$ a fixed state sequence of a HMM $\mathcal{M}$. Assume that the speech vector $\boldsymbol{o}_t$ consists of static feature vector $\boldsymbol{c}_t$ and dynamic feature vector $\Delta\boldsymbol{c}_t$, $\Delta^2\boldsymbol{c}_t$. The delta and delta-delta coefficients are computed using the following equations, where θ is set to 1 in this paper.

$$\Delta \boldsymbol{c}_t = \frac{\boldsymbol{c}_{t+\theta} - \boldsymbol{c}_{t-\theta}}{2\theta}, \Delta^2 \boldsymbol{c}_t = \frac{\Delta \boldsymbol{c}_{t+\theta} - \Delta \boldsymbol{c}_{t-\theta}}{2\theta}. \tag{1}$$

Let $\boldsymbol{W}$ be the linear transform matrix which transforms the sequence of static parameter vectors $\boldsymbol{C}$ into an ‘augmented’ sequence of static plus dynamic vectors $\boldsymbol{O}$. Then the above equations can be written as

$$\boldsymbol{O} = \boldsymbol{W}\boldsymbol{C}. \tag{2}$$

For a given state sequence S, the speech parameter generation problem is to determine the parameter sequence $\boldsymbol{C}$ which maximize $P(\boldsymbol{O}|S, \mathcal{M})$ with respect to $\boldsymbol{C}$ under the constraints (2). By setting

$$\frac{\partial \log P(\boldsymbol{W}\boldsymbol{C}|S, \mathcal{M})}{\partial \boldsymbol{C}} = \boldsymbol{0}, \tag{3}$$

a set of linear equations are obtained. The non-linear PFS control parameters used in this research are obtained by solving equation (3). The speech parameter generation technique has now been widely used in HMM-based speech synthesis systems to improve the naturalness of the synthetic speech due to the smoothing effect of the algorithm.

3. Speech synthesis experiments on TIMIT

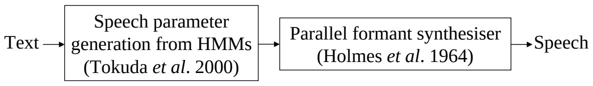

Text
Speech parameter generation from HMMs (Tokuda *et al.* 2000)
Parallel formant synthesiser (Holmes *et al.* 1964)
Speech

Figure 2: *Diagram showing the HMM-based speech synthesis system developed in this research.*

Figure 2 shows the HMM-based TTS synthesis system developed for the experimental purpose of this research. An orthographic English word string is converted into a sequence of 12 PFS control parameters based on a set of HMMs, which are trained on 12 PFS control parameters and their time derivatives. Synthesized speech is then produced by sending generated PFS control parameters to the parallel formant synthesiser.

3.1. Speech data and model set

The TIMIT corpus, downsampled to 8KHz for compatibility with the formant analyser, was used for all experiments. The training data included all male utterances from the TIMIT training set (3,252 utterances, 8 utterances discarded due to the data corruption). Each training utterance was computed using the Holmes formant analyzer to generate corresponding 12 PFS control parameters, and then converted to HTK format and augmented with Δ and Δ^2 coefficients, resulting in a representation of 36 dimensional feature vectors with a 10 ms frame rate. 49 three-state, left-to-right (no-skip) single Gaussian monophone HMMs were built on 36 PFS control parameters using HTK.

3.2. Generation of formant synthesiser control parameters

Given a set of HMMs trained on 36 PFS control parameters and a phone sequence, the synthesis problem is to generate a sequence of 12 PFS control parameters from HMMs, which are then used to drive the parallel formant synthesiser. An appropriate set of 12 PFS control parameters is crucial for the quality of the synthetic speech. [9] has shown that given appropriate synthesiser control parameters, a parallel formant synthesiser can generate extremely high quality speech which is almost indistinguishable from natural speech. The synthesis problem can be solved in two steps: 1) to decide an optimal state sequence S given a HMM model set Λ and a phone sequence $\mathcal{W}$, and 2) based on the optimal state sequence S, generate a sequence of speech parameters $\boldsymbol{o}$, which maximizes the probability $p(\boldsymbol{o}|S, \Lambda)$. For 1), if the state duration for each state is known, then the state sequence can be obtained. Hence, the problem is to determine the state duration for each state. Two methods have been used in this research to generate 12 PFS synthesiser control parameters from HMMs.

In the first approach, two assumptions are made in order to simplify the synthesis problem. Firstly, the state sequence is decided based on the expected state duration, which is determined by the self-transition probability of each state. Secondly, the state mean vectors are assumed to be the output observation vectors. The resulting synthesiser control parameters generated in this way are referred to as piecewise constant PFS control parameters.

The second approach is based on the same state sequence as above. However, the dynamic constraints imposed by the dynamic features, *i.e.*, Δ and Δ^2 coefficients, are considered and the speech parameter generation algorithm described in Section 2 is applied to generate a sequence of non-linear, smooth PFS control parameters. Speech synthesiser control parameters generated in this way are referred to as smooth trajectory PFS control parameters.

3.3. Synthetic speech evaluation methods

In order to evaluate the quality of the synthetic speech and more importantly, to assess the effect of the non-linear formant trajectories, three types of subjective listening tests were conducted to evaluate different aspects of the synthetic speech. 20 native English speakers (10 females and 10 males) were invited to participate in the test.

3.3.1. Diagnostic Rhyme Test

The Diagnostic Rhyme Test was performed to test the intelligibility of consonants in words’ initial positions. 96 word-pairs, which are taken from [10], are used in the DRT to test 6 single acoustic features (or attributes), *i.e.*, *Voicing, Nasality, Sustension, Sibilation, Graveness and Compactness*.

3.3.2. Naturalness test

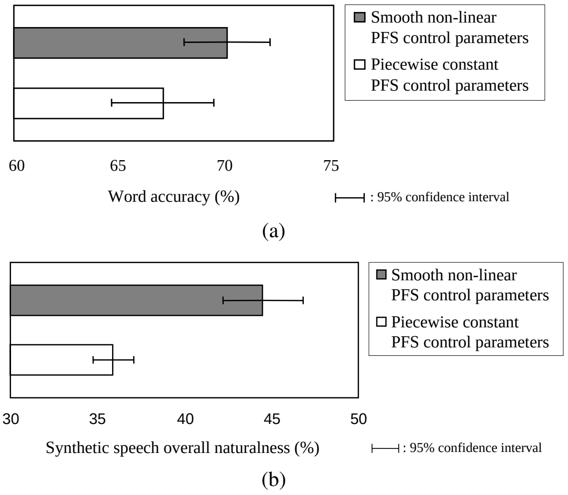
The naturalness test aims to evaluate the overall quality of the synthetic speech in sentence level. Each subject is given 20 synthetic utterances and asked to give their subjective impression on the overall quality of the speech using the mean opinion score (MOS) method, *i.e.*, '**Excellent/Good/Fair/Poor/Bad**'. Subjects are prompted to take factors into account such as naturalness, listening effort, speaking style, comprehension problems, pronunciation, speaking rate and voice pleasantness *etc.* at the beginning of the test. Each subject is given a different set of 20 synthetic utterances.

3.3.3. Mimicry Test

The purpose of the Mimicry test is to test the perceptual similarity between an individual's original speech and the synthetic speech, which is generated from a set of models adapted to that individual's speech. In other words, it tests how likely the listener thinks the two utterances are spoken by the same speaker. In this experiment, the similarity is measured in $1 - 10$ scales, where 10 means the two utterances are spoken by exactly the same speaker and 1 indicates that they are spoken by two completely different speakers. 'Semi-synthetic' (or copy synthesis) speech is used as an intermediate speech to make the comparison between original speech and synthetic speech, which is generated by passing the original speech through the Holmes formant analyzer to derive a sequence of 12 PFS control parameters, which are then sent back to the parallel formant synthesiser to generate synthetic speech.

Both MLLR and MAP adaptation techniques are used in the Mimicry test. In addition, the number of adaptation utterances varies. Since each subject speaks 10 sentences in the TIMIT corpus, the maximum number of adaptation utterances is 9 in this experiment. The number of adaptation utterances is chosen as 0 (without adaptation), 5, 9 and $\bar{9}$, where $\bar{9}$ means the model set is adapted to a different speaker using 9 adaptation utterances from that speaker.

4. Experiment results



(a)

(b)

Figure 3: *(a) DRT test results. (b) Naturalness test results.*

It can be seen from Figure 3(a) that the intelligibility of non-linear trajectory based synthetic speech is higher than synthetic speech based on piecewise constant PFS control parameters, with word accuracy percentages of 70% and 67% respectively. The naturalness test result follows the same trend, in which non-linear trajectory based synthetic speech gives a higher score, 45%, while piecewise constant control parameters based synthetic speech scores 35.9%, as is shown in Figure 3(b). A 1-tailed paired *t*-test was performed to determine if the non-linear dynamics model was effective and the results are summarized in Table 1. An alpha level of 0.05 was used for all statistical tests. Given the small values of *p*, there is strong evidence that, on average, the use of the non-linear formant trajectories does lead to improvements on intelligibility and naturalness of the synthetic speech. Moreover, a detailed analysis of the DRT results is shown in Figure 4, in which the average scores for each individual acoustic attribute are presented and comparison is made between two types of synthesiser control parameters. It can be learnt from these results that there is no benefit of using the speech parameter generation algorithm in discriminating voiced/unvoiced and nasal/oral sounds with this particular type of speech synthesis. The results for the rest of the acoustic attributes, *i.e.*, sustension, sibilation, graveness and compactness, follow the same trend, with non-linear, smooth synthetic speech giving higher scores than piecewise constant synthetic speech. In addition, a paired *t*-test discovers that only the differences between the two types of synthetic speech for attributes 'sustension' and 'compactness' are significant.

	M	*SD*	df	*t*	*p*
DRT	2.9	5.2	19	2.49	0.01
Naturalness	4.55	4.29	19	4.75	< 0.001

Table 1: *t-test results for the DRT test and naturalness test.*

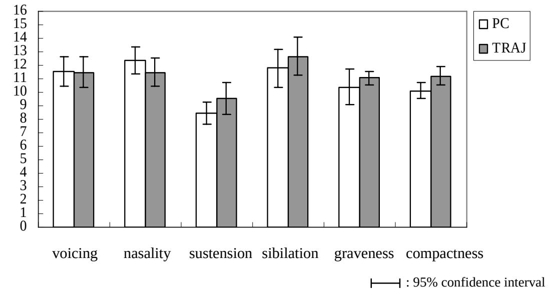
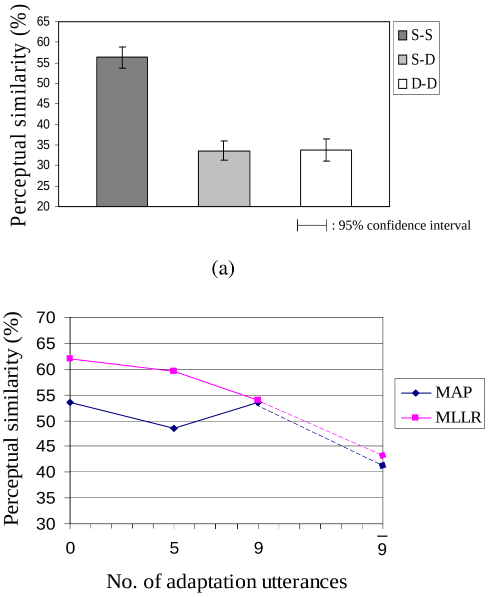


Figure 4: *DRT test results shown in individual acoustic attribute, where 'PC' stands for piecewise constant synthesiser control parameters and 'TRAJ' means non-linear, smooth synthesiser control parameters.*

The Mimicry test results are summarized in Figure 5. Figure 5(a) shows the perceptual similarity between an original utterance and a semi-synthetic speech. It can be seen that, even for the first group (S-S), *i.e.*, same speaker saying the same utterance, the similarity score between original speech and semi-synthetic speech drops to 56.8%, which suggests the semi-synthetic speech lost some information about speaker's characteristics. In addition, it can be noticed that there is no significant differences between the test results for groups S-D and D-D, with scores of 33.5% and 33.6% respectively. This indicates that the perceptual similarity between an original utterance and a semi-synthetic utterance drops sharply when the utterances are different (*i.e.*, the content of the sentence), even though they are from the same speaker.



(a)

(b)

Figure 5: *(a) Perceptual similarity between original speech and semi-synthetic speech. S-S: same speaker - same utterance; S-D: same speaker - different utterances; D-D: different speaker - different utterances. (b) Perceptual similarity between semi-synthetic speech and synthetic speech based on HMMs adapted to the same or different speaker.* $\bar{9}$*: HMMs adapted to a different speaker with 9 adaptation utterances.*

Figure 5(b) shows the perceptual similarity between semi-synthetic and synthetic speech based on HMMs adapted to the same and different speakers with different number of adaptation utterances. Number '0' means there is no adaptation, *i.e.*, speaker independent models are used to generate synthetic speech. Figure 5(b) shows that the perceptual similarity falls when the model set is adapted to a different speaker (denoted by $\bar{9}$), which is a reasonable result. However, the results of the mimicry test using MLLR and MAP adapted to the same speaker are unexpected. The perceptual similarity score decreases as the number of adaptation utterances increases when MLLR is used. When the MAP adaptation technique is used, the similarity falls from 53.5% to 48.5% as the number of adaptation utterance turns from 0 to 5, and then returns to 53.5% when 9 adaptation utterances are used.

5. Conclusions and future work

This paper shows unambiguously that the use of non-linear formant trajectories to model dynamics can improve both intelligibility and the overall quality of the HMM-based synthetic speech. The non-linear model of dynamics, generated using speech parameter generation algorithm, is particularly useful in improving the overall quality of the synthetic speech with longer contexts, *e.g.*, at sentence level, as shown in the Naturalness test. This is largely due to the smoothing effect of the algorithm, which in practice reduces the discontinuities, particularly between the boundaries of adjacent phones, which typically occur in the synthetic speech based on piecewise constant synthesiser control parameters.

In order to produce adaptable voices, conventional speaker adaptation techniques, including MLLR and MAP, were employed to generate adapted synthetic speech. However, these techniques proved to be unsuccessful in the Mimicry test. A possible explanation is that, while as few as 3 adaptation utterances can result in improvement in the recognition accuracy, the amount of the adaptation data might be too small for HMM-based speech synthesis and for this particular type of Mimicry test. However, this is due to the limitation of the TIMIT corpus. For speech recognition, MLLR is generally performs better with small amounts of adaptation data and MAP gradually catches up when more adaptation data is available. This rule of thumb seems still applicable in the similarity test, in which MLLR-based synthetic speech outperforms MAP-based synthetic speech, with at most a 10% difference between the similarity scores (when 5 adaptation utterances are used), as can be seen in Figure 5(b). Compared to other HMM-based speech synthesis systems, *e.g.*, HTS developed at Nagoya Institute of Technology [11], the overall quality of the synthetic speech is fairly poor. Although the purpose of this work is to develop dynamics model and naturalness is not the highest priority, the overall quality of the synthetic speech can surely be improved by adding prosodic structure and using context-dependent models.

Future work includes building a TTS synthesis system based on a set of multiple-level segmental HMMs, whose intermediate layer is based on 12 PFS control parameters, and generating formant synthesiser control parameters from these MSHMMs. Comparison can then be made between the synthetic speech generated by using HMMs and MSHMMs.

6. References

[1] Russell, M.J. and Jackson, P.J.B., "A multiple-level linear/linear segmental HMM with a formant-based intermediate layer", *Computer Speech and Language*, vol. 19, pp. 205–225, 2005.

[2] Holmes, J.N., Mattingly, I.G. and Shearme, J.N., "Speech synthesis by rule", *Language & Speech*, 7, pp. 127–143, 1964.

[3] Richards, H.B. and Bridle, J.S., "The HDM: a segmental Hidden Dynamic Model of coarticulation", *Proc. ICASSP*, pp. 357-360, 1999.

[4] Deng, L., "A dynamic, feature-based approach to the interface between phonology and phonetics for speech modelling and recognition", *Speech communication*, 24(4), pp. 288–323, 1998.

[5] Russell, M.J., Zheng, X. and Jackson, P.J.B., "Modelling speech signals using formant frequencies as an intermediate representation", *IET Signal Processing*, 1, pp. 43–50, 2007.

[6] Hu, H. and Russell, M.J., "Speech recognition using non-linear trajectories in a formant-based articulatory layer of a multiple-level segmental HMM", *Proc. INTERSPEECH*, pp. 2422–2425, 2008.

[7] Tokuda, K., Yoshimura, T., Masuko T., Kobayashi, T. and Kitamura T., "Speech parameter generation algorithms for HMM-based speech synthesis", *Proc. ICASSP*, vol.3, pp. 1315–1318, 2000.

[8] Acero, A., "Formant analysis and synthesis using hidden Markov models", *Proc. EUROSPEECH1999*, Budapest, 1999.

[9] Holmes, J. N., "The influence of glottal waveform on the naturalness of speech from a parallel formant synthesizer", *IEE Transactions on Audio and Electroacoustics*, 21, pp. 298–305, 1973.

[10] Pratt, R. L., "The assessment of speech intelligibility at RSRE", *Proceedings of the Institute of Acoustics*, 6, pp. 439–443, 1984.

[11] available online: http://hts.sp.nitech.ac.jp/.

Appendix D

Software Tools

D.1 Tools used in this research

A number of software tools are used in this thesis work. These include:

- SEGVit. SEGVit is a toolkit for MSHMMs training and testing. SEGVit was written by Martin Russell at the University of Birmingham in C under Linux.
- Holmes formant analyser, developed by John Holmes, which is used to automatically derive a sequence of 12PFS control parameters from a waveform file.
- Holmes formant synthesiser, developed by John Holmes. This synthesiser is used to generated synthetic speech from a sequence of 12PFS parameters.
- HTK. HTK is used for standard HMMs training, N-best list generation and evaluation.
- Matlab. Matlab is used to implement the speech parameter generation algorithm, estimate formant-to-acoustic mapping for MSHMMs, and perform statistical significance test. In addition, some of the graphs used in this thesis are produced using Matlab.
- PESQ. The reference implementation for ITU-T Recommendations P.862, P.862.1, P.862.2, Version 2.0 for PESQ is used to perform objective evaluation of the synthetic speech developed. The software is developed jointly by KPN Research, the Netherlands and British Telecommunications.

- HTML, ASP & Javascripts. These are used to build a speech synthesis evaluation system for subjective tests.
- Perl. Perl is used to deal with speech label files, textual analysis for the TTS system and file manipulation.
- Excel. Some figures in this thesis are drawn using Microsfot Excel.
- MikTex LaTeXsystem and TeXnicCenter editor are used to write this thesis under Microsoft Windows.

D.2 Softwares developed for this research

D.2.1 The speech parameter generator

This software is developed to generate a sequence of speech parameters given a state sequence and a set of HMMs using Tokuda's speech parameter generation algorithm. This program is implemented using Matlab since it involves lots of matrix operations. The pseudo code for this program is provided below:

```
START program
INITIALIZE length to zero
INITIALIZE mean to zero
INITIALIZE variance to zero
INITIALIZE count to 1
INITIALIZE template transform matrix for a single vector
SET transform matrix to template transform matrix
READ a state sequence
FOR each state in the state sequence
   CALCULATE state duration
   ADD state duration to length
   APPEND state mean to mean
   APPEND state variance to variance
```

```
ENDFOR
WHILE count < length
    SHIFT template transform matrix to the right by one column
    APPEND template transform matrix to transform matrix
    ADD 1 to count
ENDWHILE
A = transform matrix transpose * variance inverse * transform matrix
b = transform matrix transpose * variance inverse * mean transpose
COMUPTE C by solving A*C=b using QR decomposition
WRITE C as the speech parameter sequence
END program
```

D.2.2 MSHMMs *N*-best list rescoring

This program is developed to rescore an N-best list using a set of MSHMMs, which is written in C programming language, as patch code to SEGVit. The pseudo code for this program is list below:

```
START program
INITIALIZE count to zero
GET the size of the nbest list N
CREATE array of size N to hold probabilities for nbest list
INITIALIZE probability array to zero
INITIALIZE temp probability to zero
READ MSHMM model set
READ nbest list
FOR each line in nbest list
   IF line contains speech file name
      EXTRACT speech file name
      READ speech data
      READ trajectory data if doing non-linear MSHMM rescoring
   ELSEIF line equals a dot
```

```
      ADD 1 to count
      SET the count-1 element of array to temp probability
      RESET temp probability to zero
      FREE speech data
      FREE trajectory data
      IF count = N
         SEARCH the array to find the highest probability
         WRITE the index of the highest probability
         RESET all elements in probability array to zero
         RESET count=0
      ENDIF
   ELSE
      READ start time, end time, model name and state number
      FOR each model in the MSHMM model set
       IF model matches model name
          CALCULATE segment probability
          ADD segment probability to temp probability
       ENDIF
      ENDFOR
   ENDIF
ENDFOR
END program
```

List of References

Acero, A. (1999), Formant analysis and synthesis using hidden Markov models, *in* 'Proc. of the EUROSPEECH', Budapest.

Bakis, R. (1991), Coarticulation modeling with continuous-state hmms, *in* 'Proc. IEEE Workshop Automatic Speech Recognition', Arden House, New York, pp. 20–21.

Bellman, R. (2003), *Dynamic programming*, Princeton University Press. Dover paperback edition.

Bishop, C. M. (1995), *Neural Networks for Pattern Recognition*, Oxford University Press, Oxford, UK.

Black, A. W., Zen, H. & Tokuda, K. (2007), Statistical parametric speech synthesis, *in* 'Proc. of ICASSP', pp. 1229–1232.

Bourlard, H., Hermansky, H. & Morgan, N. (1996), 'Towards increasing speech recognition error rates', *Speech Commun.* **18**, 205–231.

Bourlard, H. & Morgan, N. (1994), *Connectionist Speech Recognition–A Hybrid approach*, Norwell, MA: Kluwer.

Bridle, J. (2004), Towards better understanding of the model implied by the use of dynamic features in HMMs, *in* 'Proc. of International Conference on Spoken Language Processing', Vol. 1, pp. 725–728.

Bridle, J., Deng, L., Picone, J., Richards, H., Ma, J., Kamm, T., Schuster, M., Pike, S. & Reagan, R. (1999), An investigation of segmental hidden dynamic models of speech coarticulation for automatic speech recognition, Technical report, Report of a project at the 1998 workshop on language engineering at the Center for Language and Speech Processing, The Johns Hopkins University.

Browman, C. & Goldstein, L. (1992), 'Articulatory phonology: An overview', *Phonetica* **49**, 155–180.

Chomsky, N. & Halle, M. (1968), *The Sound Pattern of English*, Harper and Row, New York.

Clark, J. & Yallop, C. (1990), *An introduction to phonetics and phonology*, Blackwell, Oxford and Cambridge, MA.

Clark, R., Richmond, K., Strom, V. & King, S. (2006), Multisyn voices for the blizzard challenge 2006, *in* 'Proc. Blizzard Workshop (Interspeech Satellite)', Pittsburgh, USA.

Cooke, N. J. (2006), Gaze-Contingent Automatic Speech Recognition, PhD thesis, University of Birmingham, UK.

Datta, R., Hu, J. & Ray, B. (2008), On efficient viterbi decoding for hidden semi-Markov models, *in* '19th International Conference on Pattern Recognition', pp. 1–4.

Davis, S. B. & Mermelstein, P. (1980), 'Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences', *IEEE Trans. ASSP* **28**(4), 357–366.

Dempster, A. P., Laird, N. M. & Rubin, D. B. (1977), 'Maximum likelihood from incomplete data via the EM algorithm', *J. Roy. Stat. Soc.* **39**(1), 1–38.

Deng, L. (1998), 'A dynamic, feature-based approach to the interface between phonology and phonetics for speech modeling and recognition', *Speech Commun.* **24**(4), 299–323.

Deng, L. (2006), *Dynamic speech models*, Morgan and Claypool Publishers.

Deng, L., Acero, A. & Bazzi, I. (2006), 'Tracking vocal tract resonances using a quantized nonlinear function embedded in a temporal constraint', *IEEE Trans. on Audio, Speech, and Language Processing* **14**(2), 425–434.

Deng, L., Aksmanovic, M., Sun, D. & Wu, J. (1994), 'Speech recognition using hidden Markov models with polynomial regression functions as nonstationary states', *IEEE Trans. on Speech and Audio Processing* **2**, 507–520.

Deng, L. & Braam, D. (1994), 'Context-dependent Markov model structured by locus equations: applications to phonetic classification', *J. Acoust. Soc. Am.* **108**(6), 3036–3048.

Deng, L. & Ma, J. (2000), 'Spontaneous speech recognition using a statistical coarticulatory model for the vocal-tract-resonance dynamics', *J. Acoust. Soc. Am.* **108**, 3036–3048.

Deng, L. & Sun, D. (1994), 'A statistical approach to automatic speech recognition usingthe atomic speech units constructed from overlapping articulatory features', *J. Acoust. Soc. Am.* **95**(5), 2702–2719.

Deng, L. & Yu, D. (2007), Use of differential cepstra as acoustic features in hidden trajectory modelling for phonetic recognition, *in* 'Proc. ICASSP', pp. 445–448.

Deng, L., Yu, D. & Acero, A. (2005), Learning statistically characterized resonance targets in a hidden trajectory model of speech coarticulation and reduction, *in* 'Proc. INTERSPEECH05', Lisbon, Portugal, pp. 1097–1100.

Deng, L., Yu, D. & Acero, A. (2006), 'Structured speech modeling', *IEEE Trans. on Audio, Speech, and Language Processing* **14**(5), 1492–1504.

Digalakis, V. (1992), Segment-based stochastic models of spectral dynamics for continuous speech recognition, PhD thesis, Boston University, MA.

Digalakis, V., Rohlicek, J. R. & Ostendorf, M. (1993), 'A dynamical system approach to continuous speech recognition', *IEEE Trans. Speech Audio Processing* **1**(4), 431–442.

Fallside, F. (1992), 'On the acquisition of speech by machines, ASM', *Speech Communication* **11**(2-3), 247–260.

Ferguson, J. D. (1980), Variable duration models for speech, *in* 'Proc. of the Symposium on the Applications of Hidden Markov Models to Text and Speech', John D. Ferguson, Ed., Princeton, NJ, IDA-CRD, pp. 143–179.

Forney, G. D. (1973), The Viterbi algorithm, *in* 'Proc. IEEE', Vol. 61, pp. 268–278.

Frankel, J. (2003), Linear dynamic models for automatic speech recognition, PhD thesis, University of Edinburgh.

Frankel, J. & King, S. (2001), ASR - articulatory speech recognition, *in* 'Proc. Eurospeech', Aalborg, Denmark, pp. 599–602.

Frankel, J. & King, S. (2007), 'Speech recognition using linear dynamic models', *IEEE Trans. on Audio, Speech, and Language Processing* **15**(1), 246–256.

Fukada, T., Tokuda, K., Kobayashi, T. & Imai, S. (1992), An adaptive algorithm for mel-cepstral analysis of speech, *in* 'Proc. ICASSP', Vol. 1, pp. 137–140.

Furui, S. (1986), 'Speaker independent isolated word recognition using dynamic features of speech spectrum', *IEEE Trans. ASSP* **34**(1), 52–59.

Furui, S. (2009), Generalization problem in ASR acoustic model training and adaptation, *in* 'IEEE Workshop on Automatic Speech Recognition and Understanding', pp. 1–10.

Gales, M. J. F. & Young, S. J. (1993), The theory of segmental hidden Markov models, Technical Report CUED/F-INFENG/TR 133.

Gao, Y., Bakis, R., Huang, J. & Zhang, B. (2000), Multistage coarticulation model combining articulatory, formant and cepstral features, *in* 'Proceedings of the International Conference on Spoken Language Proc.', Vol. 1, Beijing, pp. 25–28.

Garner, P. N. & Holmes, W. J. (1998), On the robust incorporation of formant features into hidden Markov models for automatic speech recognition, *in* 'Proceedings of the IEEE International Conference on Acoustic Speech and Signal Proc.', Vol. 1, Seattle, WA, pp. 1–4.

Garofolo, J. S., Lamel, L. F., Fisher, W. M., Fiscus, J. G., Pallett, D. S., Dahlgren, N. L. & Zue, V. (1993), *TIMIT Acoustic-Phonetic Continuous Speech Corpus*, Linguistic Data Consortium, University of Pennsylvania, Philadelphia, PA.

Gauvain, J. L. & Lee, C. (1994), Maximum a-posteriori estimation for multivariate Gaussian mixture observations of Markov chains, *in* 'IEEE Trans. on Speech and Audio Processing', Vol. 2, pp. 291–298.

Ghitza, O. & Sondhi, M. (1993), 'Hidden Markov models with templates as non-stationary states: an application to speech recognition', *Computer Speech and Language* **2**, 101–119.

Gish, H. & Ng, K. (1993), A segmental speech model with applications to word spotting, *in* 'IEEE International Conference on Acoustic, Speech and Signal Processing', Minneapolis, pp. 447–450.

Gish, H. & Ng, K. (1996), Parametric trajectory models for speech recognition, *in* 'Proc. Int. Conf. Spoken Lang. Processing', Vol. 1, pp. 466–469.

Godfrey, J. J., Holliman, E. C. & McDaniel, J. (1992), SWITCHBOARD: Telephone speech corpus for research and development, *in* 'Proc. IEEE Internat. Conf. Acoust. Speech Signal Process.', pp. 517–520.

Golub, G. H. & Kahan, W. (1965), 'Calculating the singular values and pseudo-inverse of a matrix', *Journal of the Society for Industrial and Applied Mathematics: Series B, Numerical Analysis* **2**(2), 205–224.

Golub, G. H. & Loan, C. F. V. (1996), *Matrix computations (3rd edition)*, Baltimore, Johns Hopkins.

Greenberg, C. S., Martin, A. F., Doddington, G. R. & Godfrey, J. J. (2011), Including human expertise in speaker recognition systems: report on a pilot evaluation, *in* 'Proc. ICASSP-2011'.

Gutkin, A. & King, S. (2004), Structural representation of speech for phonetic classification, *in* 'Proc. of the 17th International Conference on Pattern Recognition', Vol. 3, pp. 438–441.

Hain, T., Woodland, P. C., Evermann, G., Gales, M., Liu, X., Moore, G. L., Povey, D. & Wang, L. (2005), 'Automatic transcription of converstational telephone speech', *IEEE Transactions on Speech and Audio Processing* **13**(6), 1173–1185.

Halberstadt, A. K. & Glass, J. R. (1998), Heterogeneous measurements and multiple classifiers for speech recognition, *in* 'Proc. ICSLP', Sydney, Australia, pp. 995–998.

Haykin, S. (2001), *Kalman Filtering and Neural Networks*, Wiley Publishing.

He, X. (2001), 'A web-based intelligent tutoring system for English dictation', *International Conference on Artificial Intelligence and Computational Intelligence* **4**, 583–586.

Hermansky, H. (1990), 'Perceptual linear predictive (PLP) analysis of speech', *Journal Acoustical Soc America* **87**(4).

Hifny, Y. & Renals, S. (2009), 'Speech recognition using augmented conditional random fields', *IEEE Transactions on Audio, Speech, and Language Processing* **17**(2), 354–365.

Holmes, J. & Holmes, W. (2001), *speech synthesis and recognition*, Second edition, Taylor and Francis, London and New York.

Holmes, J. N. (1973), 'The influence of glottal waveform on the naturalness of speech from a parallel formant synthesizer', *IEE Transactions on Audio and Electroacoustics* **21**, 298–305.

Holmes, J. N. (1998), Robust measurement of fundamental frequency and degree of voicing, *in* 'Proc. Int. Conf. on Spoken Language', Sydney, Australia.

Holmes, J. N. (2001), 'Speech processing system using formant analysis', *US Patent US6292775 B1* .

Holmes, J. N., Holmes, W. J. & Garner, P. N. (1997), Using formant frequencies in speech recognition, *in* 'Proc. EUROSPEECH', Rhodes, Greece, pp. 2083–2086.

Holmes, J. N., Mattingly, I. G. & Shearme, J. N. (1964), 'Speech synthesis by rule', *Language and Speech* **7**, 127–143.

Holmes, W. (2000), *Segmental HMMs: Modelling dynamics and underlying structure for automatic speech recognition*, IMA Workshop: Mathematical Foundations of Speech Processing and Recognition. Available online: http://www.ima.umn.edu/multimedia/fall/m1.html.

Holmes, W. J. (1989), Copy synthesis of female speech using the JSRU parallel formant synthesiser, *in* 'Proc. of the EUROSPEECH'89', Paris, pp. 513–516.

Holmes, W. J. & Garner, P. N. (2000), Using formant frequencies in speech recognition, *in* 'Proceedings of the IEEE International Conference on Acoustic Speech and Signal Proc.', Vol. 3, Istanbul, Turkey, pp. 1347–1350.

Holmes, W. J. & Russell, M. J. (1995*a*), Experimental evaluation of segmental HMMs, *in* 'Proc. International Conference on Acoustics, Speech and Signal Processing', Detroit, pp. 536–539.

Holmes, W. J. & Russell, M. J. (1995*b*), Speech recognition using a linear dynamic segmental HMM, *in* 'Proc. EUROSPEECH', Madrid, pp. 1611–1614.

Holmes, W. J. & Russell, M. J. (1996), Modeling speech variability with segmental HMMs, *in* 'Proc. International Conference on Acoustics, Speech and Signal Processing', Atlanta, pp. 447–450.

Holmes, W. J. & Russell, M. J. (1997), Linear dynamic segmental HMMs: variability representation and training procedure, *in* 'Proc. International Conference on Acoustics, Speech and Signal Processing', Munich, pp. 1399–1402.

Holmes, W. & Russell, M. J. (1999), 'Probabilistic-trajectory segmental HMMs', *Comp. Speech and Lang.* **13**, 3–37.

Hu, H. & Russell, M. J. (2008), Speech recognition using non-linear trajectories in a formant-based articulatory layer of a multiple-level segmental HMM, *in* 'Proc. Interspeech', Brisbane, Australia, pp. 2422–2425.

Hu, H. & Russell, M. J. (2010), Improved modelling of speech dynamics using non-linear formant trajectories for HMM-based speech synthesis, *in* 'INTERSPEECH', Makuhari, Chiba, Japan, pp. 821–824.

Hu, Y. & Loizou, P. C. (2008), 'Evaluation of objective quality measures for speech enhancement', *IEEE Transactions on Audio, Speech, and Language Processing* **16**(1), 229–238.

Hunnicutt, A. J. & Klatt, D. (1987), *From text to speech: the MITalk system*, MIT Press, Cambridge, MA.

Hunt, A. J. & Black, A. W. (1996), Unit selection in concatenative speech synthesis system using a large speech database, *in* 'Proc. ICASSP', pp. 373–376.

Hunt, M. J. (1987), 'Delayed decisions in speech recognition-the case of formants', *Pattern Recognition Letters* **6**, 121–137.

Imai, S. (1983), Cepstral analysis synthesis on the mel frequency scale, *in* 'Proc. ICASSP', pp. 93–96.

IPA (1999), *Handbook of the International Phonetic Association*, Cambridge University Press.

Iyer, R., Gish, H., Siu, M., Zavaliagkos, G. & Matsoukas, S. (1998), Hidden Markov models for trajectory modelling, *in* 'Proc. ICSLP'.

Jackson, P. J. B., Lo, B. H. & Russell, M. J. (2002), 'Data-driven, nonlinear, formant-to-acoustic mapping for ASR', *Electron. Lett.* **38**(13), 667–669.

Jackson, P. J. B. & Russell, M. J. (2002), Models of speech dynamics in a segmental-HMM recogniser using intermediate linear representations, *in* 'Proceedings of the International Conference on Spoken Language Processing', Denver, CO, pp. 1253–1256.

Jelinek, F. (1998), *Statistical methods for speech recognition*, Cambridge, MA: MIT press.

Johnson, M. T. (2005), 'Capacity and complexity of HMM duration modelling techniques', *IEEE Signal Process. Lett.* **12**(5), 407–410.

Kalman, R. E. (1960), 'A new approach to linear filtering and prediction problems', *Transactions of the ASME-Journal of Basic Engineering* **82**(Series D), 35–45.

Kannan, A. & Ostendorf, M. (1998), 'A comparison of constrained trajectory segment models for large vocabulary speech recognition', *IEEE Transactions on Speech and Audio Processing* **6**(3), 303–306.

Kenny, P., Lennig, M. & Mermelstein, P. (1990), 'A linear predictive hmm for vector-valued observations with applications to speech recognition', *IEEE Transactions on Acoustic, Speech and Signal Processing* **38**(2), 220–225.

King, S., Frankel, J., Livescu, K., McDermott, E., Richmond, K. & Wester, M. (2007), 'Speech production knowledge in automatic speech recognition', *Journal of the Acoustical Society of America* **121**(2), 723–742.

King, S. & Karaiskos, V. (2010), The Blizzard Challenge 2010, *in* 'Proc. Blizzard Workshop'.

Lamel, L. F. & Gauvain, J. L. (1993), High performance speaker-independent phone recognition using CDHMM, *in* 'Proc. EUROSPEECH'93', pp. 121–124.

LDC (1995), *SWITCHBOARD: A User's Manual, Catalog Number LDC94S7*, Linguistic Data Consortium, University of Pennsylvania, Philadelphia, PA.

Lee, C. H. & Giachin, E. (1991), Improved acoustic modeling for speaker independent large vocabulary continuous speech recognition, *in* 'Proc. International Conference on Acoustics, Speech and Signal Processing', Vol. 1, pp. 161–164.

Lee, K.-F. (1990), 'Context-dependent phonetic hidden Markov models for speaker-independent continuous speech recognition', *IEEE Transactions on Acoustics, Speech and Signal Processing* **38**(4), 599–609.

Lee, K. & Hon, H. (1989), 'Speaker-independent phone recognition using hidden Markov models.', *IEEE Trans. on Acoustics, Speech and Signal Processing* **37**(11), 1641–1648.

Leggetter, C. & Woodland, P. C. (1995), 'Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models', *Computer Speech and Language* **9**(2), 171–185.

Levinson, S. (1986), 'Continuously variable duration hidden Markov models for automatic speech recognition', *Comput. Speech Lang.* **1**(1), 29–45.

Liporace, L. A. (1982), 'Maximum likelihood estimation for multivariate observations of Markov sources', *IEEE Trans. Information Theory* **IT-28**(5), 729–734.

Lippmann, R. P. (1997), 'Speech recognition by machines and humans', *Speech Communication* **22**(1), 1–15.

Lo, B. H. (2004), An acoustic model for speech recognition with an articulatory layer and non-linear articulatory-to-acoustic mapping, PhD thesis, University of Birmingham.

Lo, B. H. & Russell, M. J. (2003), Speech recognition using an intermediate articulatory layer and non-linear articulatory-to-acoustic mapping, *in* 'One day meeting for young speech researchers, University College London'.

Ma, J. Z. & Deng, L. (2004), 'Target-directed mixture dynamic models for spontaneous speech recognition', *IEEE Transactions on Speech and Audio Processing* **12**(1), 47–58.

Markel, J. D. & Gray, A. H. (1976), *Linear Prediction of Speech*, Berlin, Springer.

Masuko, T., Tokuda, K., Kobayashi, T. & Imai, S. (1996), Speech synthesis from HMMs using dynamic features, *in* 'Proc. of ICASSP', pp. 389–392.

Masuko, T., Tokuda, K., Kobayashi, T. & Imai, S. (1997), Voice characteristics conversion for HMM-based speehc synthesis system, *in* 'Proc. ICASSP', pp. 1611–1614.

Minami, Y., McDermott, E., Nakamura, A. & Katagiri, S. (2002), A recognition method with parametric trajectory synthesized using direct relations between static and dynamic feature vector time series, *in* 'Proc. ICASSP', Vol. 1, pp. 957–960.

Ming, J., O'Boyle, P., Owens, M. & Smith, F. J. (1999), 'A Bayesian approach for building triphone models for continuous speech recognition', *IEEE Transactions on Speech and Audio Processing* **7**(6), 678–684.

Ming, J. & Smith, F. J. (1998), Improved phone recognition using Bayesian triphone models, *in* 'Proc. ICASSP', pp. 409–412.

Mitchell, C., Harper, M. & Jamieson, L. (1995), 'On the complexity of explicit duration HMM's', *IEEE Transactions on Speech and Audio Processing* **3**(3), 213–217.

Mohamed, A., Dahl, G. E. & Hinton, G. (2012), 'Acoustic modeling using deep belief networks', *IEEE Transactions on Audio, Speech, and Language Processing* **20**(1), 14–22.

Moore, B. C. J., Glasberg, B. R. & Vickers, D. A. (1999), 'Further evaluation of a model of loudness perception applied to cochlear hearing loss', *J. Acoust. Soc. Am.* **106**(2), 898–907.

O'Shaughnessy, D. (1987), *Speech communication: human and machine*, Addison-Wesley.

Ostendorf, M. (1991), Integration of diverse recognition methodologies through reevaluation of n-best sentence hypotheses, *in* 'Proc. DARPA Speech and Natural Language Processing Workshop', pp. 83–87.

Ostendorf, M. (1999), Moving beyond the beads-on-a-string model of speech, *in* 'IEEE Workshop on Automatic speech recognition and Understanding', Vol. 1, Keystone, pp. 79–83.

Ostendorf, M., Digalakis, V. V. & Kimball, O. A. (1996), 'From HMM's to segment models: a unified view of stochastic modeling for speech recognition', *IEEE Trans. on Speech and Audio Process.* **4**(5), 360–378.

Ostendorf, M., Kannan, A., Kimball, O. & Rohlicek, J. R. (1992), Continuous word recognition based on the stochastic segment model, *in* 'Proc. DARPA Workshop CSR'.

Ostendorf, M. & Roukos, S. (1989), 'A stochastic segment model for phoneme-based continuous speech recognition', *IEEE Trans. Acoust., Speech, Signal Processing* **37**(12), 1857–1869.

Oura, K., Zen, H., Nankaku, Y., Lee, A. & Tokuda, K. (2006), Hidden semi-Markov model based speech recognition system using weighted finite-state transducer, *in* 'Proc. IEEE International Conference on Acoustics, Speech and Signal Processing', Vol. 1, pp. 33–36.

Paul, D. & Baker, J. (1992), The design for the Wall Street Journal based CSR corpus, *in* 'Proc. DARPA Speech and Natural Language Workshop', Morgan Kaufmann, Austin, TX, pp. 357–360.

Peterson, G. E. & Barney, H. L. (1952), 'Control methods used in a study of the vowels', *J. Acoust. Soc. Am.* **24**(2), 175–184.

Picone, J., Pike, S., Reagan, R., Kamm, T., Bridle, J., Deng, L., Ma, Z., Richards, H. & Schuster, M. (1999), Initial evaluation of hidden dynamic models on conversational speech, *in* 'Proc. ICASSP', Vol. 1, pp. 109–112.

Pratt, R. L. (1984), The assessment of speech intelligibility at RSRE, *in* 'Proceedings of the Institute of Acoustics', Vol. 6, pp. 439–443.

Rabiner, L. R. (1989), A tutorial on hidden Markov models and selected applications in speech recognition, *in* 'Proc. IEEE', Vol. 77, pp. 257–286.

Richards, H. B. & Bridle, J. S. (1999), The HDM: a segmental hidden dynamic model of coarticulation, *in* 'Proc. ICASSP', Phoenix, AZ, pp. 357–360.

Richmond, K. (2001), Estimating Articulatory Parameters from the Acoustic Speech Signal, PhD thesis, Centre for Speech Technology Research, Edinburgh University.

Rix, A. W., Beerends, J. G., Hollier, M. P. & Hekstra, A. P. (2001), Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs, *in* 'Proc. ICASSP 2001', Vol. 2, pp. 749–752.

Robinson, A. (1994), 'An application of recurrent nets to phone probability estimation', *IEEE Trans. on Neural Networks* **5**(2), 298–305.

Rosti, A.-V. I. (2004), Linear Gaussian models for speech recognition, PhD thesis, Cambridge University.

Roweis, S. & Ghahramani, Z. (1999), 'A unifying review of linear Gaussian models', *Neural Computation* **11**(2).

Russell, M. J. (1993), A segmental HMM for speech pattern modelling, *in* 'Proc. IEEE International Conference on Acoustics, Speech and Signal Processing', Minneapolis, pp. 499–502.

Russell, M. J. (1997), Progress towards speech models that model speech, *in* 'Proc. IEEE Workshop on Automatic Speech Recognition and Understanding', Santa Barbara, CA, pp. 115–123.

Russell, M. J. (2003), *Spoken language processing*, Lecture notes, version 11, The University of Birmingham.

Russell, M. J. (2004), *A Unified Model for Speech Recognition and Synthesis*, University of Birmingham.

Russell, M. J. (2005), 'Reducing computational load in segmental HMM decoding for speech recognition', *Electron. Lett.* **41**(25), 1408–1409.

Russell, M. J. (2012), Personal communication.

Russell, M. J. & Holmes, W. J. (1997), 'Linear trajectory segmental HMMs', *IEEE Signal Processing Letters* **4**, 72–74.

Russell, M. J., Moore, R. K., Tomlinson, M. J. & Deacon, J. C. A. (1983), RSRE speech database recordings 1983: Part II. Recordings made for automatic speech recognition assessment and research, Technical Report RSRE Report 84008, Malvern, UK.

Russell, M. J., Zheng, X. & Jackson, P. J. B. (2007), 'Modelling speech signals using formant frequencies as an intermediate representation', *IET Signal Processing* **1**, 43–50.

Russell, M. & Jackson, P. J. B. (2005), 'A multiple-level linear/linear segmental HMM with a formant-based intermediate layer', *Comp. Speech and Lang.* **19**(2), 205–225.

Russell, M. & Moore, R. (1985), Explicit modelling of state occupancy in hidden Markov models for automatic speech recognition, *in* 'Proc. ICASSP'85', pp. 5–8.

Seide, F., Zhou, J. & Deng, L. (2003), Coarticulation modeling by embedding a target-directed hidden trajectory model into HMM-MAP decoding and evaluation, *in* 'Proc. ICASSP', pp. 748–751.

Soong, F. K. & Rosenberg, A. E. (1988), 'On the use of instantaneous and transitional spectral information in speaker recognition', *IEEE Transactions on Acoustics, Speech and Signal Processing* **36**(6), 871–879.

Tamura, M., Masuko, T., Tokuda, K. & Kobayashi, T. (1998), Speaker adaptation for HMM-based speech synthesis system using MLLR, *in* 'Proc. ESCA/COCOSDA Third International Workshop on Speech synthesis', pp. 273–276.

Toda, T. & Tokuda, K. (2001), Speech parameter generation algorithm considering global variance for HMM-based speech synthesis, *in* 'EUROSPEECH', pp. 2801–2804.

Tokuda, K., Kobayashi, T. & Imai, S. (1995), Speech parameter generation from HMM using dynamic features, *in* 'Proc. ICASSP', pp. 660–663.

Tokuda, K., Masuko, T., Miyazaki, N. & Kobayashi, T. (1999), Hidden Markov models based on multi-space probability distribution for pitch pattern modeling, *in* 'Proc. ICASSP'99', pp. 229–232.

Tokuda, K., Masuko, T., Yamada, T., Kobayashi, T. & Imai, S. (1995), An algorithm for speech parameter generation from continuous mixture HMMs with dynamic features, *in* 'Proc. EUROSPEECH', pp. 757–760.

Tokuda, K., Yoshimura, T., Masuko, T., Kobayashi, T. & Kitamura, T. (2000), Speech parameter generation algorithms for HMM-based speech synthesis, *in* 'Proc. ICASSP', Vol. 3, pp. 1315–1318.

Tokuda, K., Zen, H. & Black, A. (2002), An HMM-based speech synthesis system applied to English, *in* 'IEEE Speech Synthesis Workshop'.

Tokuda, K., Zen, H. & Kitamura, T. (2003), Trajectory modelling based on HMMs with the explicit relationship between static and dynamic features, *in* 'Proc. Eurospeech 2003', Geneva, Switzerland.

Valentini-Botinhao, C., Yamagishi, J. & King, S. (2011), Evaluation of objective measures for intelligibility prediction of HMM-based synthetic speech in noise, *in* 'Proc. ICASSP', pp. 5112–5115.

Viterbi, A. J. (1967), 'Error bounds for convolutional codes and an asymptotically optimal decoding algorithm', *IEEE Trans. Informat. Theory* **IT-13**, 260–269.

Voiers, W. D. (1977), 'Diagnostic evaluation of speech intelligibility', *Benchmark papers in acoustics* **11**. Speech intelligibility and speaker recognition (M. Hawley, ed.) Dowden, Hutinson and Ross.

Wellekens, C. J. (1987), Explicit time correlation in hidden Markov models for speech recognition, *in* ‘Proc. ICASSP’, pp. 384–387.

Wilkinson, N. & Russell, M. J. (2002), Improved phone recognition on TIMIT using formant frequency data and confidence measures, *in* ‘Proceedings of the IEEE International Conference on Acoustic Speech and Signal Proc.’, Denver, CO, pp. 2121–2142.

Williams, C. (2005), ‘How to pretend that correlated variables are independent by using difference observations’, *Neural Computation* **17**, 1–6.

Wilpon, J. G. & Rabiner, L. R. (1985), ‘A modified k-means clustering algorithm for use in speaker-independent isolated word recognition’, *IEEE Transactions on ASSP* **33**, 587–594.

Wilpon, J., Lee, C. H. & Rabiner, L. (1991), Improvements in connected digit recognition using higher order spectral and energy features, *in* ‘Proc. International Conference on Acoustics, Speech and Signal Processing’, Vol. 1, pp. 349–352.

Woodland, P. C., Leggetter, C. J., Odell, J. J., Valtchev, V. & Young, S. J. (1995), The 1994 HTK large vocabulary speech recognition system, *in* ‘Proc. IEEE International Conference on Acoustics, Speech and Signal Processing’, Detroit, pp. 73–76.

Wrench, A. A. (2001), A new resource for production modelling in speech technology, *in* ‘Proc. Workshop on Innovations in Speech Processing’.

Yamagishi, J. (2006), Average-Voice-Based Speech Synthesis, PhD thesis, Tokyo Institute of Technology.

Yamagishi, J. & Kobayashi, T. (2007*a*), ‘Average-voice-based speech synthesis using HSMM-based speaker adaptation and adaptive training’, *IEICE Trans. Information and Systems* **E90-D(2)**, 533–543.

Yamagishi, J. & Kobayashi, T. (2007*b*), ‘Average-voice-based speech synthesis using HSMM-based speaker adaptation and adaptive training’, *IEICE Trans. Inf. Syst.* **E90-D**(2), 533–543.

Yamagishi, J., Kobayashi, T., Nakano, Y., Ogata, K. & Isogai, J. (2009), ‘Analysis of speaker adaptation algorithms for HMM-based speech synthesis and a constrained SMAPLR adaptation algorithm’, *IEEE Transactions on Audio, Speech, and Language Processing* **17**(1), 66–83.

Yamagishi, J., Kobayashi, T., Tachibana, M., Ogata, K. & Nakano, Y. (2007), Model adaptation approach to speech synthesis with diverse voices and styles, *in* ‘Proc. ICASSP’, pp. 1233–1236.

Yamagishi, J., Ogata, K., Nakano, Y., Isogai, J. & Kobayashi, T. (2006), HSMM-based model adaptation algorithms for average-voice-based speech synthesis, *in* ‘Proc. ICASSP’, pp. 77–80.

Yoshimura, T. (2002), Simultaneous modeling of phonetic and prosodic parameters, and characteristic conversion for HMM-based text-to-speech systems, PhD thesis, Department of Electrical and Computer Engineering, Nagoya Institute of Technology.

Yoshimura, T., Tokuda, K., Masuko, T., Kobayashi, T. & Kitamura, T. (1998), Duration modelling in HMM-based speech synthesis system, *in* 'Proc. of ICSLP', Vol. 2, pp. 29–32.

Yoshimura, T., Tokuda, K., Masuko, T., Kobayashi, T. & Kitamura, T. (1999), Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis, *in* 'Proc. EUROSPEECH', Vol. 5, pp. 2347–2350.

Young, S., Evermann, G., Gales, M., Hain, T., Kershaw, D., Moore, G., Odell, J., Ollason, D., Povey, D., Valtchev, V. & Woodland, P. (2005), *The HTK Book (for HTK Version 3.3)*, Cambridge University.

Young, S. J. (1995), Large vocabulary continuous speech recognition: A review, *in* 'Proc. IEEE Workshop on Automatic Speech Recognition and Understanding', Snowbird, Utah, pp. 3–28.

Young, S. J., Odell, J. J. & Woodland, P. C. (1994), Tree-based state tying for high accuracy acoustic modelling, *in* 'Proc. Human Language Technology Workshop', Plainsboro NJ, Morgan Kaufman Publishers Inc, pp. 307–312.

Young, S. J., Woodland, P. C. & Byrne, W. J. (1994), 'Spontaneous speech recognition for the credit card corpus using the HTK toolkit', *IEEE Trans. on Audio and Speech Processing* **2**(4), 615–621.

Yu, S.-Z. & Kobayashi, H. (2003), 'An efficient forward-backward algorithm for an explicit-duration hidden Markov model', *IEEE Signal Process. Lett.* **10**(1), 11–14.

Yun, Y.-S. & Oh, Y.-H. (2002), 'A segmental-feature HMM for continuous speech recognition based on a parametric trajectory model', *Speech Communication* **38**, 115–130.

Zen, H., Tokuda, K. & Kitamura, T. (2007), 'Reformulating the HMM as a trajectory model by imposing explicit relationships between static and dynamic feature vector sequences', *Comp. Speech and Lang.* **21**, 153–173.

Zhang, L. (2009), Modelling Speech Dynamics with Trajectory-HMM, PhD thesis, University of Edinburgh, UK.

Zhou, J., Seide, F. & Deng, L. (2003), Coarticulation modeling by embedding a target-directed hidden trajectory model into HMM-modeling and training, *in* 'Proceedings of the IEEE International Conference on Acoustic Speech and Signal Proc.', Vol. 1, Hong Kong, pp. 744–747.

Zue, V. W. (1991), *Notes on speech spectrogram reading*, Course notes, MIT, Cambridge, MA.

Zwicker, E. & Fastl, H. (1999), *Psychoacoustics-Facts and Models*, Springer, Berlin.