# Efficient Decoding Strategies for Conversational Speech Recognition Using a Constrained Nonlinear State-Space Model

Jeff Z. Ma and Li Deng, *Senior Member, IEEE*

***Abstract*—In this paper, we present two efficient strategies for likelihood computation and decoding in a continuous speech recognizer using an underlying nonlinear state-space dynamic model for the hidden speech dynamics. The state-space model has been specially constructed so as to be suitable for the conversational or casual style of speech where phonetic reduction abounds. Two specific decoding algorithms, based on optimal state-sequence estimation for the nonlinear state-space model, are derived, implemented, and evaluated. They successfully overcome the exponential growth in the original search paths by using the path-merging approaches derived from Bayes' rule. We have tested and compared the two algorithms using the speech data from the Switchboard corpus, confirming their effectiveness. Conversational speech recognition experiments using the Switchboard corpus further demonstrated that the use of the new decoding strategies is capable of reducing the recognizer's word error rate compared with two baseline recognizers, including the HMM system and the nonlinear state-space model using the HMM-produced phonetic boundaries, under identical test conditions.**

***Index Terms*—Bayes rule, decoding, Kalman filter, path merging, state-space model.**

## I. Introduction

INHERENT difficulties of the popular hidden Markov model (HMM) approach to recognition of naturally-speaking speech have been known for sometime (e.g., [2], [4], [7], [8], etc.), so are the limitations of some more recent approaches based on the stochastic segment models (SSM) [3], [9], [10], [14], [19], [20], [26], [27], [33] that have largely ignored correlations across pronunciation units. Along the SSM direction, we have in recent years been developing a new approach to conversational speech recognition which directly addresses the following key issue: how to represent in the speech model the phenomenon of phonetic reduction that is ubiquitous in conversational speech and how to represent the associated speaking rate variation that is scaled in a continuous fashion? Aimed at explicit incorporation of the mechanisms of phonetic reduction, several versions of the global, "super-segmental" speech models described in [10]–[12], [28] have, in various ways, integrated the speaking rate variation with the contextual variation in spontaneous speech. They have provided explicit correlation across pronunciation units at the levels internal to the direct acoustic observation. The mathematical structure of such novel speech models is a highly constrained, nonlinear state-space system, with the model parameters switching from one set of values to another at a switching rate corresponding to either the feature-sized, phone-sized, or syllable-sized phonological units of speech. In this paper, we confine the speech model to a specific version where the continuous state variables are partially hidden vocal-tract resonances (VTRs), and where the switching rate of model parameters are based on the phone-sized speech unit.

The VTR-based, nonlinear state-space model described in this paper can be considered as generalizations of the HMM and SSM. It is more complex so that the speech dynamics can be more properly handled. The conceptual and theoretical similarities and differences between the HMM/SSM and the nonlinear state-space model can be summaried as follows:[1] 1) they are similar left-to-right statistical models, with trainable model parameters; 2) they are intended to model the speech feature trajectories, with different degrees of precision; 3) unlike the HMM and SSM, the nonlinear state-space model has an internal representation of the production-affiliated speech dynamics. This new dynamics has the special target-directed property, and the property of continuity across speech units, which are not shared with the HMM/SSM; and 4) they have very different complexities in decoding, which we address below.

One major limitation of SSMs for continuous speech recognition is the decoding complexity, which forces many SSMs to adopt the re-scoring strategy for evaluation. The decoding difficulty is even greater for our long-span or "super-segmental" dynamic model. Because of the explicit continuity constraint and correlation imposed across the hidden dynamics of a sequence of the phone units,[2] the decoding methods are substantially more complex than the well-established decoding methods developed for the HMMs [21] and than the existing decoding methods for the stochastic segment models [13], [27]. Hence, one main challenge for developing successful strategies for conversational speech recognition

Manuscript received June 29, 1999; revised September 24, 2002. A compressed version of this paper was presented at Eurospeech 2001. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Bryan George.

J. Z. Ma was with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada N2L 3G1. He is now with BBN Technologies, Cambridge, MA 02138 USA.

L. Deng was with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada N2L 3G1. He is now with Microsoft Research, Redmond, WA 98052-6399 USA (e-mail: deng@microsoft.com).

Digital Object Identifier 10.1109/TSA.2003.818075

[1]Some detailed discussions can be found in [12]

[2]Similar continuity constraints were applied to the dynamic models in [15], [29]. No such constraint and correlation have been imposed on the HMMs and on any previous versions of the stochastic segment models.

using the "super-segmental" dynamic model is to develop efficient decoding strategies. The subject matter of this paper is the formulation of the decoding problem associated with the constrained state-space model that explicitly represents cross-phone correlation and phonetic reduction via the use of partially hidden VTR state variables.

The work in this paper has been partly motivated by some earlier work on the various versions of the switching state-space model developed in control engineering, time series analysis, neural computation, and in econometrics [6], [16]–[18], [22], [31], [32], [35], [36]. Common to those diverse fields is various forms of the switching model, or a model with switching parameters, where the model parameters change among a set of different parameters and where the change is allowed to occur at any time. A linear case of the switching model is

$$Z(k) = F[s(k)]Z(k-1) + W(k-1, s(k)) \quad (1)$$

$$O(k) = H[s(k)]Z(k) + V(k, s(k)) \quad (2)$$

where $s(k)$ denotes the mode of the model at time $k$, which is assumed to be one of $M$ possible modes, $s(k) \in \{s_i\}_{i=1}^{M}$. Such earlier work typically dealt with several simplified cases of the model, and focused only on the state estimation and not the decoding (i.e., state-sequence estimation) problem. In [32], the switching is refined only to the measurement equation (to trace the trajectories of multiple targets). In [22], the state estimation problem for linear state-space models was considered, where the switching was allowed for both state and measurement equations (to trace the recession and booming stages in the economy). The model in [18] allowed the parameter switching to occur in the error covariances as well. A more general algorithm for the state estimation was given in [6], where Bayes' rule was used when the parameter switching of the state-space model follows an arbitrary process including the Markovian one. A comprehensive review on the use of switching models for the multiple-target tracking problems has been provided in [30], [31]. A detailed review of the application of the switching models to econometrics has been provided in [35].

Moving beyond the original applications in target tracking and in econometrics, the research reported in this paper represents our novel contribution of applying the specially constructed switching state-space model to functional modeling of speech production and to speech recognition. The speech production process can be well fitted into the switching state-space model since each phone in a finite number of phones in a language can be associated with a largely distinct target vocal-tract shape and its related acoustic resonance structure. When a speech utterance is produced, the vocal tract shape or resonance (continuous state in the model) changes relatively smoothly from one target phone to another, where the target shapes determined by the model parameters are made to switch from one target phone to its temporally adjacent one.

A further contribution of the research described in this paper is to extend the solutions to the state estimation problem to those of the state *sequence* estimation (i.e., decoding) problem. All the earlier work on the switching model surveyed above focused on the state estimation problem where the source of the difficulty is exponential growth in the number of switching combinations over time. The earlier solutions to the state estimation problem have first been rederived and modified to suit the special constraints associated with the speech model. These solutions have then been extended to those to the decoding problem used for speech recognition.

This paper is organized as follows. In Section II, the switching state-space model for the dynamics of VTRs is reviewed in order to set up the context in which the formulation of the decoding strategy is established as the problem of optimal state sequence estimation. Details of the statistical state estimation, including all main steps of derivation and two types of approximate solutions to the state estimation problem, are contained in Section III. Armed with the approximate solutions to the state estimation problem, a one-pass dynamic-programming-based decoding algorithm is developed and presented in Section IV. This algorithm aims at finding the globally optimal path under the constraints imposed on the switching process of the model. The decoding strategies described in Section IV are evaluated on a conversational speech recognition task defined from the Switchboard data. Details of the evaluation experiments and the improved performance using an N-best re-scoring paradigm are reported in Section V. Finally, we draw conclusions and discuss main contributions of this work in Section VI.

## II. Constrained Switching State-Space Model for Speech Dynamics

The constrained, nonlinear, and switching state-space model for the target-directed dynamics of VTRs is reviewed in this section, for the purpose of setting up the context where the Bayesian formulation of the decoding strategy will be established in Sections III–V of this paper. Details about this model can be found in [12].

The mathematical structure of the model is described by the following coupled state equation and measurement equation:

$$Z(k) = \Phi^{(j)} Z(k-1) + \left(I - \Phi^{(j)}\right) T^{(j)} + W^{(j)}(k) \quad (3)$$

$$O(k) = h(Z(k)) + V^{(j)}(k). \quad (4)$$

The state equation (3) describes the constrained linear dynamics of VTRs, in which the state variable $Z(k)$ represents vector-valued VTR frequencies (with dimensionality of four in this work: $F_1, F_2, F_3$, and $F_4$), $T^{(j)}$ and $\Phi^{(j)}$ are the vector-valued VTR target and "time constant" (diagonal) matrix, respectively. (They are both automatically estimated using the EM algorithm as presented in [12].) $W^{(j)}(k)$ is an additive, zero-mean, Gaussian i.i.d. "state" noise with covariance matrix $Q^{(j)}$. The most significant property of the state equation (3) is the constraint on the parameters which forces the state variable $Z(k)$ to exhibit the asymptotic behavior directed temporally toward the target $T^{(j)}$.

The measurement equation (4) describes the noisy nonlinear relationship between the state vector, $Z(k)$, and the direct acoustic observation, $O(k)$. Examples of the acoustic observation are the Mel frequency cepstral coefficient (MFCC) measurements computed from a conventional speech preprocessor. $V^{(j)}(k)$ is the additive "observation" noise modeled also by a zero-mean, Gaussian i.i.d. process with the covariance

matrix $R^{(j)}$. The multivariate nonlinear mapping, $h(Z)$, is implemented by a global multi-layer perceptron (MLP) as presented in this paper, whose Jacobian matrix can be analytically evaluated as needed for the state estimation problem to be discussed in Section III.

In the model formulation of (3) and (4), $j$ is the index of the switching dynamic regime which corresponds to a unique set of model parameters. The switching model parameters are discrete at the dynamic regime boundaries (i.e., no constraints), but the underlying continuous state is *constrained* to be continuous at the dynamic regime boundaries. (This constraint is implemented in the state estimation algorithm described in Section III.) In the work presented in this paper, each dynamic regime occupies a segment of speech roughly of the size of a surface form of a phone. That is, the parameters of the model are phone dependent, and because of the incorporation of contextual modeling via target-directed dynamics *and* via the continuity constraint, the parameters of the model are *not* made conditioned on context-dependent phones such as triphones.

In short, the model parameter set consists of $\Theta = \{\Phi, T, R, Q\}$ plus the parameters contained in the nonlinear function $h(\cdot)$ (e.g., MLP weights). As the speech utterance traverses from left to right in time, phone-sized dynamic regimes switch from one to another, which induces the switching process among $M$ parameter sets $\Theta^{(j)} = \{\Phi^{(j)}, T^{(j)}, R^{(j)}, Q^{(j)}\}$, where $j = 1, 2, \ldots, M$ and $M$ is the total number of phones.

The dynamic speech model reviewed above which underlies our new recognizer has been developed based on new principles of modeling hidden dynamics of speech production. In summary, this speech model, as constructed for the VTR dynamics employed in this work, is formulated in mathematical terms as a constrained and simplified nonlinear dynamic system with switching parameters. This is a special version of the general statistical hidden dynamic model described in [10], [11], where more detailed motivations and mathematical construction for the dynamic speech models of such a type are available.

Given a sequence of observations $O_1^K = \{O(1), O(2), \ldots, O(K)\}$, and assuming that this entire sequence is generated by model $j$ (i.e., no model switching occurs during the generation of the observation sequence), then the log-likelihood of the new model in (3) and (4) can be computed according to

$$\begin{aligned} L\left(O_1^K|\Theta\right) &= \log p\left(O(1), O(2), \ldots, O(K)|\Theta^{(j)}\right) \\ &= \sum_{k=1}^{k} \log p\left(O(k)|O_1^{k-1}, \Theta^{(j)}\right) \\ &= -\frac{1}{2}\sum_{k=1}^{K}\left\{\left|\Sigma_{\tilde{O}_k}^{(j)}\right| + \left(\tilde{O}_k^{(j)}\right)'\left[\Sigma_{\tilde{O}_k}^{(j)}\right]^{-1}\tilde{O}_k^{(j)}\right\} \\ &\quad + const. \end{aligned} \tag{5}$$

where we assume $p(O(1)) = p(O(1)|O(0))$, and define $\tilde{O}_k^{(j)} = O(k) - E\left[O(k)|O_1^{k-1}, \Theta^{(j)}\right]$, which we call the innovation sequence. $\Sigma_{\tilde{O}_k^{(j)}}$ is the covariance of $p\left(O(k)|O_1^{k-1}, \Theta^{(j)}\right)$ or the covariance of $\tilde{O}_k^{(j)}$. $\tilde{O}_k^{(j)}$ and $\Sigma_{\tilde{O}_k}^{(j)}$ are calculated from the Kalman filtering procedure. The Kalman filter converts the likelihood of the entire observation sequence (which is temporally correlated) into the summation of likelihoods at local time points (uncorrelated innovations). Under the approximate Gaussian assumption for the innovations, the minimum mean square error criterion for state estimation becomes that of maximum likelihood for each local time point. Therefore, when the model is allowed to switch, the only problem that remains is how to determine the switching sequence (or phone sequence), using maximum likelihood as the criterion.

Kalman filter [23], [25] is a recursive state estimation algorithm. For the model in (3) and (4), the extended Kalman filter (EKF) algorithm that we use in this work and that handles the nonlinear function $h$ is given in

$$\hat{Z}_{k|k-1}^{(j)} = \Phi^{(j)}\hat{Z}_{k-1|k-1}^{(j)} + \left(I - \Phi^{(j)}\right)T^{(j)} \tag{6}$$

$$\Sigma_{k|k-1}^{(j)} = \Phi^{(j)}\Sigma_{k-1|k-1}^{(j)}\Phi^{(j)'} + Q^{(j)} \tag{7}$$

$$\tilde{O}_k^{(j)} = O(k) - h\left[\hat{Z}_{k|k-1}^{(j)}\right] \tag{8}$$

$$\Sigma_{\tilde{O}_k}^{(j)} = H_{k|k-1}\Sigma_{k|k-1}^{(j)}\left(H_{k|k-1}\right)' + R^{(j)} \tag{9}$$

$$K_k^{(j)} = \Sigma_{k|k-1}^{(j)}\left(H_{k|k-1}\right)'\left[\Sigma_{\tilde{O}_k}^{(j)}\right]^{-1} \tag{10}$$

$$\hat{Z}_{k|k}^{(j)} = \hat{Z}_{k|k-1}^{(j)} + K_k^{(j)}\tilde{O}_k^{(j)} \tag{11}$$

$$\Sigma_{k|k}^{(j)} = \Sigma_{k|k-1}^{(j)} - K_k^{(j)}\Sigma_{\tilde{O}_k}^{(j)}\left(K_k^{(j)}\right)' \tag{12}$$

where $H_{k|k-1}$ is the Jacobian matrix of $h(\cdot)$ at point $\hat{Z}_{k|k-1}^{(j)}$. Recursion at $k$ is initialized by the filtered values at recursion $k-1$, $\hat{Z}_{k-1|k-1}^{(j)}$ and $\Sigma_{k-1|k-1}^{(j)}$. This makes the calculation at each local time point depend on the entire past history.

If the model switches among $M$ different modes (or phones), the switching combinations will grow exponentially. On the other hand, the local score computation for our new model depends on the past history. Therefore, no path deletion is theoretically allowed during the decoding (detailed analyses can be found in [24]), which makes the search for the new model difficult. In the following, we will first review how the switching state estimation problem with the exponential growth has been handled. We will then exploit the relevant ideas to deal with our discrete-state decoding problem.

## III. Formulation of the Decoding Problem: State Estimation

For the conventional state-space model with *fixed* parameters, the goal of optimal (in both the Bayesian sense [34] and the minimum mean square error sense [25]) state estimation (filtering) is to calculate the conditional mean and covariance of the hidden state $Z(k)$ given the observations up to time $k$. Let us use $\hat{Z}_{k|k}$ to denote the conditional mean, and $\Sigma_{k|k}$ the conditional covariance, then

$$\hat{Z}_{k|k} = E\left[Z(k)|O_1^k\right]$$

and

$$\Sigma_{k|k} = Cov\left[Z(k)|O_1^k\right].$$

Generalizing from the fixed-parameter case to the case with switching parameters like the model given in (1) and (2), the

mean and covariance of $Z(k)$ will be conditioned not only on the observations up to time $k$ but also on the evolution history of the model switching. Therefore, the conditional mean and covariance become

$$E\left[Z(k)|s_k = i_k, s_{k-1} = i_{k-1}, \ldots, s_1 = i_1, O_1^k\right] \quad (13)$$

and

$$Cov\left[Z(k)|s_k = i_k, s_{k-1} = i_{k-1}, \ldots, s_1 = i_1, O_1^k\right] \quad (14)$$

where $s_k$ represents the *discrete state*[3] at time frame $k$, $i_k$ $(1 \leq i_k \leq M)$ is the index to the phone which the dynamics in the system switches to at time $k$. In the remaining of this paper, we refer to (13) and (14) as the elemental estimator in the overall state estimation procedure to be discussed.

If the switching evolution process, $s_k = i_k, s_{k-1} = i_{k-1}, \ldots, s_1 = i_1$, were known, the conventional state estimation techniques would directly apply [1], [25]. However, in speech recognition decoding problems of concern, the switching evolution process is unknown. In principle, all the possibilities (paths) of the evolution have to be considered during the estimation. At each time point, any one of the $M$ models could be chosen, and such as, there are potentially as many as $M^k$ paths for the switching evolution up to time $k$. Each of these paths forms an elemental estimator based on the conventional state estimation, and hence there are a prohibitively large number $M^k$ of the elemental estimators at time $k$. How to obtain the overall state estimate from the huge number of elemental estimators in a computationally efficient manner? The Bayes theory holds the key to the solution.

For simplicity purposes, let us first adopt the following notations:

$\Psi_k$ all paths of the switching evolution up to time $k$ ($M^k$ in total);

$\psi_k(n)$ $n$th path of $\Psi_k$. This can be explicitly written as

$$\psi_k(n) = \left\{s_{n,k} = i_{n,k}, s_{n,k-1} = i_{n,k-1}, \ldots, s_{n,1} = i_{n,1}\right\} = \left\{s_{n,k} = i_{n,k}, \psi_{k-1}(m)\right\}$$

where $\psi_{k-1}(m)$ is the $m$th path of $\Psi_{k-1}$, from which $\psi_k(n)$ stems. That is, $\psi_{k-1}(m) = \{s_{m,k-1} = i_{m,k-1} = i_{n,k-1}, \ldots, s_{m,1} = i_{m,1} = i_{n,1}\}$.

$P_k(n)$ probability of $\psi_k(n)$ being true given $O_1^k$; that is, $P_k(n) = Pr(\psi_k(n)|O_1^k)$.

Using the above notations, the following state estimation algorithm, by applications of Bayes' rule, is obtained (see the derivation in Appendix).

1) Elemental estimator for the fixed, $n$th path

$$\hat{Z}_{n,k|k} = E\left[Z(k)|s_{n,k} = i_{n,k}, s_{n,k-1} = i_{n,k-1}, \ldots, s_{n,1} = i_{n,1}, O_1^k\right] \quad (15)$$

$$\Sigma_{n,k|k} = Cov\left[Z(k)|s_{n,k} = i_{n,k}, s_{n,k-1} = i_{n,k-1}, \ldots, s_{n,1} = i_{n,1}, O_1^k\right]. \quad (16)$$

[3]In this work, each discrete state corresponds to a distinct phone or model with phone-dependent model parameters.

2) Estimator for the state (continuous) and its covariance

$$\hat{Z}_{k|k} = \sum_{n \in \Psi_k} P_k(n) \hat{Z}_{n,k|k} \quad (17)$$

$$\Sigma_{k|k} = \sum_{n \in \Psi_k} P_k(n) \times \left\{\Sigma_{n,k|k} + \left[\hat{Z}_{n,k|k} - \hat{Z}_{k|k}\right]\left[\hat{Z}_{n,k|k} - \hat{Z}_{k|k}\right]'\right\}. \quad (18)$$

3) Recursive update of $P_k(n)$

$$P_k(n) = \frac{p\left(O(k)|\psi_k(n), O_1^{k-1}\right)}{p\left(O(k)|O_1^{k-1}\right)} \times P\left(S_{n,k} = i_{n,k}|\psi_{k-1}(m), O_1^{k-1}\right) P_{k-1}(m) \quad (19)$$

where the normalizing factor $p(O(k)|O_1^{k-1})$ is computed from

$$\sum_{n \in \Psi_k} p\left(O(k)|\psi_k(n), O_1^{k-1}\right) \times P\left(S_{n,k} = i_{n,k}|\psi_{k-1}(m), O_1^{k-1}\right) P_{k-1}(m) \quad (20)$$

in which $p(O(k)|\psi_k(n), O_1^{k-1})$ is calculated during the elemental estimation, and $P(S_{n,k} = i_{n,k}|\psi_{k-1}(m), O_1^{k-1})$ characterizes the evolution of the switching, which is typically assumed to be known in advance.

The state in the switching dynamic system can be recursively estimated by the above algorithm. However, the number of the elemental estimators grows exponentially with time. This gives rise to a great computational difficulty in the implementation of the algorithm [6].

To overcome such a difficulty, let us assume that the model switching follows a first-order Markov chain. The first-order Markov chain can be expressed by the following transition probability matrix:

$$P = \begin{pmatrix} P_{11} & P_{12} & \cdots & P_{1M} \\ P_{21} & P_{22} & \cdots & P_{2M} \\ \vdots & \vdots & \vdots & \vdots \\ P_{M1} & P_{M2} & \cdots & P_{MM} \end{pmatrix} \quad (21)$$

where $P_{ij} = P(S_k = j|S_{k-1} = i)$ and $\sum_{j=1}^{M} P_{ij} = 1$ for all $i$. $P(S_k = j|S_{k-1} = i)$ is the probability of the system dynamics which switches from discrete state (i.e., phone) $i$ to discrete state $j$ at time $k$.

For this Markov chain, the current switching depends only on the previous one,so $P(S_{n,k} = i_{n,k}|\psi_{k-1}(m), O_1^{k-1})$ becomes $P(S_{n,k} = i_{n,k}|S_{n,k-1} = i_{n,k-1})$. Therefore, (19) can be drastically simplified to

$$P_k(n) = \frac{p\left(O(k)|\psi_k(n), O_1^{k-1}\right)}{p\left(O(k)|O_1^{k-1}\right)} \times P\left(S_{n,k} = i_{n,k}|S_{n,k-1} = i_{n,k-1}\right) P_{k-1}(m). \quad (22)$$

However, even under this Markovian assumption, the elemental estimator is still conditioned on the entire switching

history, $\psi_k(n)$. As such, the number of the elemental estimators remains exponential growth with respect to time. To make the state-estimation algorithm computationally feasible, approximations have to be made which would make the estimation suboptimal as a price paid for alleviating computation burden. We have explored two types of approximations, both based on merging the path hypotheses in a time-synchronous manner. The two ways of path merging and the associated suboptimal state estimation algorithms are discussed below.

### A. Approximation I: Generalized Pseudo-Bayesian Approach

The generalized Pseudo-Bayesian approach (GPB) has been used for state estimation in the target-tracking and econometrics literature for general dynamic systems [6], [17], [22], [31], [35]. We extended this approach, which has been developed specifically for speech recognition applications, to the specially constrained, nonlinear dynamic system model described in Section II for the speech dynamics.

First-order GPB and second-order GPB for general linear dynamic systems have been available in the literature [31]. In the first-order GPB (GPB1), the state estimate is carried out under each possible current model at each time $k$. In the switching evolution history $\psi_k(n) = \{s_{n,k} = i_{n,k}, s_{n,k-1} = i_{n,k-1}, \ldots, s_{n,1} = i_{n,1}\}$, only the most recent term $s_{n,k} = i_{n,k}$ is kept, while the other terms are dropped. That is, $\psi_k(n)$ is approximated by $\{s_{n,k} = i_{n,k}\}$. Therefore, only $M$ out of $M^k$ paths are considered by merging all "older" paths for all models at time $k-1$. This approach is very efficient in computation but we judge that the price for loss of accuracy would be too great to be used for implementing our speech recognition decoding algorithms.

In the second-order GPB (GPB2) which we use for implementing one of the speech recognition decoding algorithms, the state estimation is carried out under each possible pair of current and previous model at each time $k$. In the switching evolution history, $\psi_k(n) = \{s_{n,k} = i_{n,k}, s_{n,k-1} = i_{n,k-1}, \ldots, s_{n,1} = i_{n,1}\}$, the most recent two terms, $s_{n,k} = i_{n,k}$ and $s_{n,k-1} = i_{n,k-1}$, are considered and earlier terms are dropped. That is, $\psi_k(n)$ is approximated by $\{s_{n,k} = i_{n,k}, s_{n,k-1} = i_{n,k-1}\}$. Therefore, a total of $M^2$ paths for the path combination are considered at each time frame. For this case, all the paths for each model at time $k-1$ are merged. The merging in GPB2 is not as heavy as GPB1 and its estimation accuracy is substantially higher. We describe and derive GPB2 in detail for the constrained, nonlinear switching dynamic model of speech reviewed in Section II.

Given the GPB2 approximation, the precise elemental estimator (15) and (16) has now been approximated by

$$E[Z(k)|s_{n,k} = i_{n,k}, s_{n,k-1} = i_{n,k-1}, O_1^k]$$

and

$$Cov[Z(k)|s_{n,k} = i_{n,k}, s_{n,k-1} = i_{n,k-1}, O_1^k].$$

For simplicity purposes, in the following derivation of the state estimate, we will use $s_k$, $s_{k-1}$, $j$ and $i$ to represent $s_{n,k}$, $s_{n,k-1}$, $i_{n,k}$ and $i_{n,k-1}$, respectively , and will adopt the following notation:

$$\begin{aligned}
\hat{Z}_{k|k-1}^{(i,j)} &= E\left[Z(k)|s_k = j, s_{k-1} = i, O_1^{k-1}\right] \\
&\quad (elemental\ predictor) \\
\Sigma_{k|k-1}^{(i,j)} &= Cov\left[Z(k)|s_k = j, s_{k-1} = i, O_1^{k-1}\right] \\
\hat{Z}_{k|k}^{(i,j)} &= E\left[Z(k)|s_k = j, s_{k-1} = i, O_1^{k}\right] \\
&\quad (elemental\ filter) \\
\Sigma_{k|k}^{(i,j)} &= Cov\left[Z(k)|s_k = j, s_{k-1} = i, O_1^{k}\right] \\
\hat{Z}_{k|k}^{(j)} &= E\left[Z(k)|s_k = j, O_1^{k}\right] \\
&\quad (merged\ state\ estimate) \\
\Sigma_{k|k}^{(j)} &= Cov\left[Z(k)|s_k = j, O_1^{k}\right].
\end{aligned}$$

*1) Merging and State Estimation:* With use of the above notation, the state estimate, (17) and (18), has become

$$\hat{Z}_{k|k} = \sum_{j=1}^{M}\sum_{i=1}^{M} P\left(s_k = j, s_{k-1} = i|O_1^k\right)\hat{Z}_{k|k}^{(i,j)} \tag{23}$$

$$\begin{aligned}
\Sigma_{k|k} = &\sum_{j=1}^{M}\sum_{i=1}^{M} P\left(s_k = j, s_{k-1} = i|O_1^k\right) \\
&\times \left\{\Sigma_{k|k}^{(i,j)} + \left[\hat{Z}_{k|k}^{(i,j)} - \hat{Z}_{k|k}\right] \right. \\
&\left. \times \left[\hat{Z}_{k|k}^{(i,j)} - \hat{Z}_{k|k}\right]'\right\}.
\end{aligned} \tag{24}$$

In GPB2, the merged estimate at node $j$ is calculated according to

$$\begin{aligned}
\hat{Z}_{k|k}^{(j)} &= E\left[Z(k)|s_k = j, O_1^k\right] \\
&= \sum_{i=1}^{M} P\left(s_{k-1} = i|s_k = j, O_1^k\right) \\
&\quad \times E\left[Z(k)|s_k = j, s_{k-1} = i, O_1^k\right] \\
&= \sum_{i=1}^{M} P\left(s_{k-1} = i|s_k = j, O_1^k\right)\hat{Z}_{k|k}^{(i,j)}
\end{aligned} \tag{25}$$

$$\begin{aligned}
\Sigma_{k|k}^{(j)} &= Cov\left[Z(k)|s_k = j, O_1^k\right] \\
&= \sum_{i=1}^{M} P\left(s_{k-1} = i|s_k = j, O_1^k\right) \\
&\quad \times \left\{Cov\left[Z(k)|s_k = j, s_{k-1} = i, O_1^k\right] \right. \\
&\qquad + \left(E\left[Z(k)|s_k = j, s_{k-1} = i, O_1^k\right]\right. \\
&\qquad \left. - E\left[Z(k)|s_k = j, O_1^k\right]\right) \\
&\qquad \cdot \left(E\left[Z(k)|s_k = j, s_{k-1} = i, O_1^k\right]\right. \\
&\qquad \left. \left. - E\left[Z(k)|s_k = j, O_1^k\right]\right)'\right\} \\
&= \sum_{i=1}^{M} P\left(s_{k-1} = i|s_k = j, O_1^k\right) \\
&\quad \times \left\{\Sigma_{k|k}^{(i,j)} + \left(\hat{Z}_{k|k}^{(i,j)} - \hat{Z}_{k|k}^{(j)}\right)\left(\hat{Z}_{k|k}^{(i,j)} - \hat{Z}_{k|k}^{(j)}\right)'\right\}.
\end{aligned} \tag{26}$$

That is, the merged quantities are obtained by summing up all the possibilities at each model according to their individual posterior probabilities, $P(s_{k-1}=i|s_k=j,O_1^k)$, which is also called merging probability. It is easy to show that the state estimate, (23) and (24), is equivalent to

$$\hat{Z}_{k|k}=\sum_{j=1}^{M}P\left(s_k=j|O_1^k\right)\hat{Z}_{k|k}^{(j)} \tag{27}$$

$$\Sigma_{k|k}=\sum_{j=1}^{M}P\left(s_k=j|O_1^k\right)\times\left\{\Sigma_{k|k}^{(j)}+\left[\hat{Z}_{k|k}^{(j)}-\hat{Z}_{k|k}\right]\left[\hat{Z}_{k|k}^{(j)}-\hat{Z}_{k|k}\right]'\right\}. \tag{28}$$

The posterior probabilities, $P(s_{k-1}=i|s_k=j,O_1^k)$ and $P(s_k=j|O_1^k)$ (also called model probability), are calculated recursively. According to Bayes rule and by straightforward conditional probability manipulation, they are equal to

$$P(s_{k-1}=i|s_k=j,O_1^k)=\frac{p\left(O(k),s_k=j,s_{k-1}=i|O_1^{k-1}\right)}{\sum_{i=1}^{M}p\left(O(k),s_k=j,s_{k-1}=i|O_1^{k-1}\right)} \tag{29}$$

$$P(s_k=j|O_1^k)=\frac{\sum_{i=1}^{M}p\left(O(k),s_k=j,s_{k-1}=i|O_1^{k-1}\right)}{\sum_{j=1}^{M}\sum_{i=1}^{M}p\left(O(k),s_k=j,s_{k-1}=i|O_1^{k-1}\right)}. \tag{30}$$

In (29) and (30), $p(O(k),s_k=j,s_{k-1}=i|O_1^{k-1})$ is computed by

$$\begin{aligned}p(O(k),s_k=j,s_{k-1}=i|O_1^{k-1})&=p(O(k)|s_k=j,s_{k-1}=i,O_1^{k-1})\\&\quad\times P(s_k=j|s_{k-1}=i,O_1^{k-1})\\&\quad\times P(s_{k-1}=i|O_1^{k-1})\end{aligned} \tag{31}$$

where $p(O(k)|s_k=j,s_{k-1}=i,O_1^{k-1})$ is obtained during the elemental estimation, $P(s_k=j|s_{k-1}=i,O_1^{k-1})=p(s_k=j|s_{k-1}=i)$ is the transition probability and is known,[4] and $P(s_{k-1}=i|O_1^{k-1})$ is recursively calculated according to (30) and (31) with the initial value given at time 0.

*2) Elemental Estimator:* Many possible approaches are available for elemental estimation (i.e., conditional state estimation given the path history). The simplest approach we have adopted in the current recognizer implementation is the Extended Kalman Filter (EKF) algorithm.[5] In our recognizer implementation, the EKF algorithm ( [1], [23], [25], [30], [34]) has been adapted to suit the special structure of the speech model (Section II) incorporating target-directed constraint and cross-regime continuity constraint. For the elemental estimator in GPB2, the EKF algorithm given in (6)–(12) is tailored to

$$\hat{Z}_{k|k-1}^{(i,j)}=\Phi^{(j)}\hat{Z}_{k-1|k-1}^{(i)}+\left(I-\Phi^{(j)}\right)T^{(j)} \tag{32}$$

$$\Sigma_{k|k-1}^{(i,j)}=\Phi^{(j)}\Sigma_{k-1|k-1}^{(i)}\Phi^{(j)\prime}+Q^{(j)} \tag{33}$$

$$\tilde{O}_k^{(i,j)}=O(k)-h\left[\hat{Z}_{k|k-1}^{(i,j)}\right] \tag{34}$$

$$\Sigma_{\tilde{O}_k}^{(i,j)}=H_{k|k-1}\Sigma_{k|k-1}^{(i,j)}\left(H_{k|k-1}\right)'+R^{(j)} \tag{35}$$

$$K_k^{(i,j)}=\Sigma_{k|k-1}^{(i,j)}\left(H_{k|k-1}\right)\left(\Sigma_{\tilde{O}_k}^{(i,j)}\right)^{-1} \tag{36}$$

$$\hat{Z}_{k|k}^{(i,j)}=\hat{Z}_{k|k-1}^{(i,j)}+K_k^{(i,j)}\tilde{O}_k^{(i,j)} \tag{37}$$

$$\Sigma_{k|k}^{(i,j)}=\Sigma_{k|k-1}^{(i,j)}-K_k^{(i,j)}\Sigma_{\tilde{O}_k}^{(i,j)}\left(K_k^{(i,j)}\right)' \tag{38}$$

where $H_{k|k-1}$ is the Jacobian matrix of nonlinear function $h(\cdot)$ at point $\hat{Z}_{k|k-1}^{(i,j)}$. In the speech recognizer described in this paper, a three-layered MLP is used for the nonlinear function $h(\cdot)$. The Jacobian matrix of the MLP is

$$H=\begin{bmatrix}\frac{\partial h_1}{\partial x_1}&\frac{\partial h_1}{\partial x_2}&\cdots&\frac{\partial h_1}{\partial x_L}\\\frac{\partial h_2}{\partial x_1}&\frac{\partial h_2}{\partial x_2}&\cdots&\frac{\partial h_2}{\partial x_L}\\\vdots&\vdots&&\vdots\\\frac{\partial h_I}{\partial x_1}&\frac{\partial h_I}{\partial x_2}&\cdots&\frac{\partial h_I}{\partial x_L}\end{bmatrix} \tag{39}$$

where $\partial h_i/\partial x_l$ ( $1\le i\le I$ and $1\le l\le L$, $I$ is the dimension of output and $L$ is the dimension of input) is equal to

$$\frac{\partial h_i}{\partial x_l}=\sum_{n=1}^{N}\left\{W_{in}\cdot g\left(\sum_{l=1}^{L}w_{nl}x_l\right)\times\left(1-g\left(\sum_{l=1}^{L}w_{nl}x_l\right)\right)\cdot w_{nl}\right\} \tag{40}$$

in which $N$ is the number of nodes in the hidden layer, $W_{in}$ are weights connecting output and hidden layers, $w_{nl}$ weights connecting hidden and input layers, and $g(\cdot)$ is the sigmoid function $g(x)=1/1+\exp(-x)$ with its derivative $g(x)(1-g(x))$.

Note that the current elemental estimator is initialized by the merged state estimate of model $i$ at the previous time, $\hat{Z}_{k-1|k-1}^{(i)}$ and $\Sigma_{k-1|k-1}^{(i)}$.

While using the EKF for the elemental estimator implementation, an assumption is made that $p(O(k)|s_k=j,s_{k-1}=i,O_1^{k-1})$ in (31) is Gaussian and computed by

$$\begin{aligned}p\left(O(k)|s_k=j,s_{k-1}=i,O_1^{k-1}\right)&=(2\pi)^{-I/2}\left|\Sigma_{\tilde{O}_k}^{(i,j)}\right|^{-1/2}\\&\quad\times\exp\left\{-\frac{1}{2}\left(\tilde{O}_k^{(i,j)}\right)'\left[\Sigma_{\tilde{O}_k}^{(i,j)}\right]^{-1}\tilde{O}_k^{(i,j)}\right\}.\end{aligned} \tag{41}$$

*3) Algorithm Summary:* Each recursion of the GPB2 algorithm, tailored for our specially constrained, nonlinear

[4]This is the transition probability of the discrete Markov switching process. In our phone-based model construction, this probability gives the bi-phone "language model," which we fixed to be 0.5 in our recognizer implementation.

[5]Despite the linearity in state equation (3) used in our speech model of Section II, a nonlinear filtering algorithm is needed due to the nonlinear measurement (4) in the model.

switching state-state model of speech, as detailed above in this subsection can be summarized below for each time step.

1. Calculate elemental estimates, $\hat{Z}_{k|k}^{(i,j)}$ and $\Sigma_{k|k}^{(i,j)}$, for $i,j = 1,2,\ldots,M$, using the EKF listed in (32)–(38), using the merged state estimates at the previous time step, $\hat{Z}_{k-1|k-1}^{(i)}$ and $\Sigma_{k-1|k-1}^{(i)}$ as initial values.
2. Calculate the merging probabilities, $p(s_{k-1} = i|s_k = j, O_1^k)$, according to (29) and (30) and (41).
3. Merge states for each model at the current time step using the merging probabilities according to (25) and (26). (These will be used for the next time-step recursion).
4. Calculate model probabilities, $P(s_k = j|O_1^k)$, according to (30), (31) and (41).
5. Calculate the state estimate, $\hat{Z}_{k|k}$ and $\Sigma_{k|k}$, according to (27) and (28).

### B. Approximation II: Interacting Multiple Model Approach

In the general Interacting Multiple Model (IMM) approach [30], [31], the state estimate is carried out under each possible current model at each time step $k$. However, each possibility has its own initial value obtained by a weighted sum over all the estimates at the previous time $k-1$. This differs from the GPB1 approach [31] in which all possibilities are initialized by the same value. The merging in the IMM approach takes place just before the estimation begins at each current model, which differs from GPB2 in which the merging takes place after the estimation is completed at each current model. Like GPB1, the IMM approach has the same $M$ possibilities considered at each time step, and hence have the same computational complexity. IMM is more accurate than GPB1. Compared with GPB2, the IMM approach has significantly lower computational complexity (a factor of $M$), but since the merging is more aggressive, it is expected to have a lower estimation accuracy.

*1) Merging and State Estimation:* In this subsection, we adapt the general IMM state estimation approach to suit the special need of our specially constrained, nonlinear state-space model of speech dynamics. In this approach, the elemental estimator is approximated by $E[Z(k)|s_k = j, O_1^k]$ and $Cov[Z(k)|s_k = j, O_1^k]$. The EKF algorithm for the elemental estimator in IMM is the same as that given in (6)–(12), except that the initialization values, $\hat{Z}_{k-1|k-1}^{(j)}$ and $\Sigma_{k-1|k-1}^{(j)}$, are replaced by special ones, $\bar{Z}_{k-1|k-1}^{(j)}$ and $\bar{\Sigma}_{k-1|k-1}^{(j)}$. The new initial values are obtained by merging according to

$$\begin{aligned}\bar{Z}_{k-1|k-1}^{(j)} \equiv& E\left[Z(k-1)|s_k = j, O_1^{k-1}\right] \\ =& \sum_{i=1}^{M} P\left(s_{k-1} = i|s_k = j, O_1^{k-1}\right) \\ & \times \hat{Z}_{k-1|k-1}^{(i)} \qquad (42)\end{aligned}$$

$$\begin{aligned}\bar{\Sigma}_{k-1|k-1}^{(j)} =& \sum_{i=1}^{M} P\left(s_{k-1} = i|s_k = j, O_1^{k-1}\right) \\ & \times \left\{\Sigma_{k-1|k-1}^{(i)} + \left(\hat{Z}_{k-1|k-1}^{(i)} - \bar{Z}_{k-1|k-1}^{(j)}\right)\right. \\ & \left. \times \left(\hat{Z}_{k-1|k-1}^{(i)} - \bar{Z}_{k-1|k-1}^{(j)}\right)'\right\}. \qquad (43)\end{aligned}$$

In (42) and (43), $P(s_{k-1} = i|s_k = j, O_1^{k-1})$ is called mixing probability, and is computed recursively from Bayes' rule

$$\begin{aligned}P&\left(s_{k-1} = i|s_k = j, O_1^{k-1}\right) \\ &= \frac{P\left(s_k = j|s_{k-1} = i, O_1^{k-1}\right) P\left(s_{k-1} = i|O_1^{k-1}\right)}{\sum_{i=1}^{M} P\left(s_k = j|s_{k-1} = i, O_1^{k-1}\right) P\left(s_{k-1} = i|O_1^{k-1}\right)} \\ &= \frac{P\left(s_k = j|s_{k-1} = i\right) P\left(s_{k-1} = i|O_1^{k-1}\right)}{\sum_{i=1}^{M} P\left(s_k = j|s_{k-1} = i\right) P\left(s_{k-1} = i|O_1^{k-1}\right)} \qquad (44)\end{aligned}$$

where the model probability $P(s_{k-1} = j|O_1^{k-1})$ is updated as in GPB2 (given by (30)). However, the calculation of $p(O(k)|s_k = j, s_{k-1} = i, O_1^{k-1})$ has been changed to

$$\begin{aligned}p&\left(O(k)|s_k = j, s_{k-1} = i, O_1^{k-1}\right) \\ &= (2\pi)^{-I/2}\left|\Sigma_{\tilde{O}_k}^{(j)}\right|^{-1/2} \\ &\quad \times \exp\left\{-\frac{1}{2}\left(\tilde{O}_k^{(j)}\right)'\left[\Sigma_{\tilde{O}_k}^{(j)}\right]^{-1}\tilde{O}_k^{(j)}\right\}. \qquad (45)\end{aligned}$$

*2) Algorithm Summary:* The entire IMM algorithm can be summarized as follows (each time step).

1. Calculate mixing probabilities, $P(s_{k-1} = i|s_k = j, O_1^{k-1})$, using (44).
2. Calculate merged initial conditions, $\bar{Z}_{k-1|k-1}^{(j)}$ and $\bar{\Sigma}_{k-1|k-1}^{(j)}$, for each current model according to (42) and (43), given the estimates at the previous time step $k-1$.
3. Calculate the estimates, $\hat{Z}_{k|k}^{(j)}$ and $\Sigma_{k|k}^{(j)}$, by running the EKF listed in (6)–(12) with the special initial values obtained in Step 2.
4. Update model probabilities, $P(s_k = j|O_1^k)$, according to (30), (31) and (45).
5. Calculate the state and its covariance estimates, $\hat{Z}_{k|k}$ and $\Sigma_{k|k}$, according to (27) and (28).

### C. Analysis and Comparison of GPB2 and IMM Approximations

As demonstrated in Sections III-A and B, the GPB2 approach takes into account all pairs of model switchings from time $k-1$ to $k$, each pair has its initial value from its previous model. The IMM approach considers only the models at the current time $k$, but each model has its initial value obtained by a special way averaging over the state estimates of the models at the previous time $k-1$. In GPB2 the merging takes place after the elemental estimation (EKF). In contrast, the merging happens before the elemental estimation in IMM. A graphical comparison of GPB2 and IMM merging strategies is illustrated in Fig. 1, where for convenience $Z$ and $V$ are used to represent $\hat{Z}$ and $\Sigma$, respectively).

While on the surface, it may appear that the difference of the GPB2 and IMM approximations lies only in the order of merging and of elemental estimator (EKF), the deep-seated difference lies in their distinct ways of approximating a mixture Gaussian density. This is elaborated below. The estimation in (25) and (26) is eventually equal to the computation of pdf

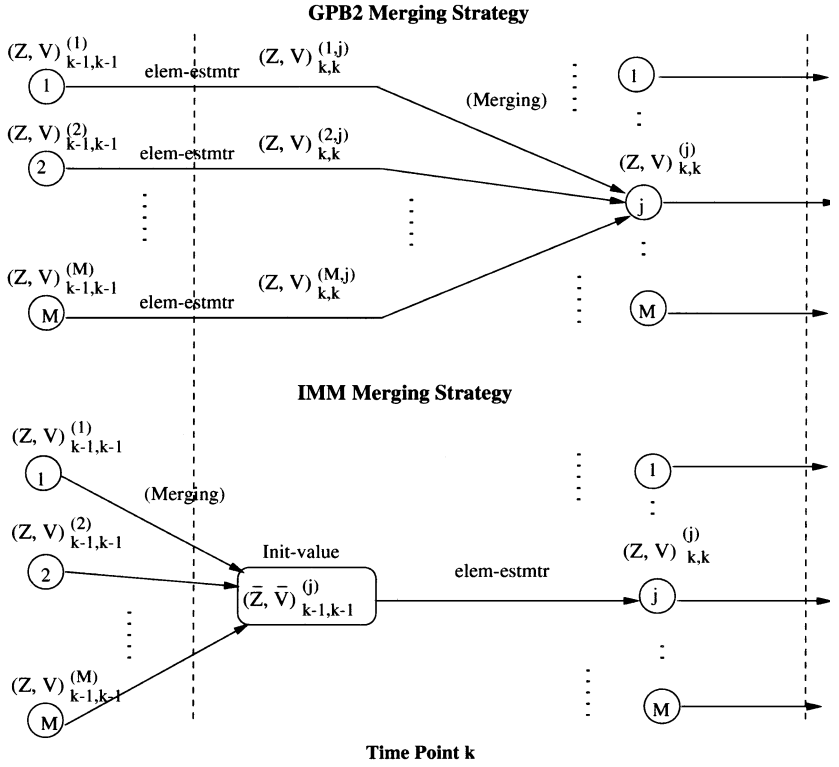


Fig. 1. GPB2 and IMM merging Strategies.

$p(Z(k)|s_k = j, O_1^k)$. The posterior pdf $p(Z(k)|s_k = j, O_1^k)$ can be shown to be

$$p\left(Z(k)|s_k = j, O_1^k\right) = \frac{p\left(O(k)|s_k = j, Z(k)\right)}{p\left(O(k)|s_k = j, O_1^{k-1}\right)} \times p\left(Z(k)|s_k = j, O_1^{k-1}\right) \quad (46)$$

where $p(Z(k)|s_k = j, O_1^{k-1})$ can be expressed as (see proof in the Appendix)

$$p\left(Z(k)|s_k = j, O_1^{k-1}\right) = \int p(Z(k)|Z(k-1), s_k = j, O_1^{k-1}) \times \left[\sum_{i=1}^{M} p\left(Z(k-1)|s_{k-1} = i, O_1^{k-1}\right) \times P\left(s_{k-1} = i|s_k = j, O_1^{k-1}\right)\right] dZ(k-1). \quad (47)$$

Let us examine (46) and (47) in detail. We first note that the component in the integrand of (47), $\sum_{i=1}^{M} p(Z(k-1)|s_{k-1} = i, O_1^{k-1})P(s_{k-1} = i|s_k = j, O_1^{k-1})$, is a Gaussian sum. This makes the entire integral of (47) to be also a Gaussian sum. Also note that the denominator in (46) is a constant independent of $Z(k)$ and the numerator, $p(O(k)|s_k = j, Z(k), O_1^{k-1})$, is a Gaussian. This further makes $p(Z(k)|s_k = j, O_1^k)$ in (46) to be a Gaussian sum.

The essence of the GPB2 and IMM approximations, both of which alleviate computational problems in state estimation, is the use of approximations of a Gaussian sum by a unimodal Gaussian; that is, the Gaussian sum, $p(Z(k)|s_k = j, O_1^k)$, is approximated by a unimodal Gaussian. They use such approximations in different ways, resulting in different approximation accuracy with computational tradeoff. In the IMM approximation, the Gaussian sum in (47), $\sum_{i=1}^{M} p(Z(k-1)|s_{k-1} = i, O_1^{k-1})P(s_{k-1} = i|s_k = j, O_1^{k-1})$, is approximated by a single Gaussian before the integration (or the EKF process), which results in the approximate unimodal Gaussian for $p(Z(k)|s_k = j, O_1^k)$. In contrast, in the GPB2 the Gaussian sum, $p(Z(k)|s_k = j, O_1^k)$, is approximated directly by a single Gaussian after the integration. Therefore, the GPB2 have to carry out $M$ integrations (one for each mixture component) and the IMM only carries out one integration. This shows that IMM is $M$ times more efficient than GPB2.

## IV. Decoding Strategies Using Approximate State Estimation

The state estimation methods described in Section V have been adapted from the methods well established in control engineering, econometrics, and time series analysis, where the interest has been mainly focused on the accuracy of the (continuous) state estimation at local times. Our contribution therein has been to tailor the methods to suit our special speech model's structure and to generate the maximum likelihood for our models. In this section, we incorporate these methods into the decoding strategy for our new models, aiming to search (time-synchronous) for the *global* optimal path through all discrete states (i.e., the entire parameter switching history). Once the global path is found, the recognizer produces the text output according to this global path. Before we present the two global decoding algorithms (incorporating the GPB2 and IMM, respectively), we briefly explain why a straightforward use of the state estimation methods described in Section III is *not* desirable. For both the GPB2 and the IMM, at each time step the posterior probability, $P(s_k = j|O_1^k)$, can be computed

```
For each frame, O(k), in the observation sequence of an utterance ( 0 <k<K+1)
   Initialize Best_score(k) to be negative infinity.
   For each node (model or phone) j in the lattice (or n-best list) (0 < j < M+1)
      Initialize max. path log-likelihood L(k,j) to be negative infinity.
      Run step 1,2,3,4 of IMM state estimation algorithm to obtain local score L(O(k) | j )
        and other updates for the next time step.
      For each node i which can enter into node j
         Let path log-likelihood L(k,j,i ) = L(k-1, i) + L(O(k) | j ).
         if ( L(k,j,i ) > L(k, j ) )
            L(k,j ) = L(k,j,i ),
            Remeber the tracking-back pointer from node j to i.
         endif
         if ( L(k,j ) > Best_score(k) )
            Best_score(k) = L(k,j );
         endif
      End
   End
   Impose boundary constraints here if needed.
   Do path pruning here if needed.
     [ if ( | (L(k,n) - Best_score(k))/ Best_score(k) | > Pruning_thres ), delete node n
       for next expansion ( 0 < n < M+1).
End
Find the path with the highest likelihood.
Backtrack to obtain segment boundaries if needed.
```

Fig. 2. One-pass dynamic programming decoding algorithm with the incorporation of the IMM merging strategy.

for each discrete state $j$ or phone (30). Therefore, at each time step, if we were to "decode" the discrete state based on the highest posterior probability $(\arg\max_{1\le j\le M} P(s_k = j|O_1^k))$, we would have a global "maximum a posterior" path. However, the posterior probability $P(s_k = j|O_1^k)$ is just conditioned on the observations up to time $k$ rather than on the whole sequence $O_1^K$, so the "maximum a posterior" path is not consistent with the normal decoding criterion of maximizing the joint likelihood of observation sequence and path: $\max_{\psi_K\in\Psi_K} l(O_1^K, \psi_K)$ $(\propto l(\psi_K|O_1^K))$ [6]. On the other hand, in the general switching models there are no constraints on the model switching, but in speech there are always some constraints on phone model or word model switching, such as the left-to-right constraint.

To implement a desirable decoding rule, $\max_{\psi_K\in\Psi_K} l(O_1^K, \psi_K)$, we have developed a new dynamic programming based strategy. For the new speech dynamic model given in (3) and (4), the standard Viterbi algorithm used for HMM cannot be applied directly to search for optimal dynamic regimes. This is so because different paths entering one node $j$ in the trellis bring different initial values (due to different path histories) to the EKF and score calculation at node $j$. For example, in (32)–(38), the filtered values and score calculation at state $j$ and time $k$ of the path coming from state $i$ depend on the initial values $\hat{Z}^{(i)}_{k-1|k-1}$ and $\Sigma^{(i)}_{k-1|k-1}$. These differential initial values will produce different filtered values and scores, which will be again used, due to the dynamics' continuity imposed across the adjacent discrete states, as initial values for the expansion of those paths at the next time $k+1$ (details can be found in [24]).

However, the GPB2 and IMM algorithms developed in Section III have provided effective ways to prevent the above problem of exponential growth of paths. The GPB2 uses (25) and (26) to merge the different filtered values of those paths entering node $j$ at time $k$. $\hat{Z}^{(i,j)}_{k|k}$ and $\Sigma^{(i,j)}_{k|k}$ are merged to a single point, $\hat{Z}^{(j)}_{k|k}$ and $\Sigma^{(j)}_{k|k}$, according to their a-posteriori probabilities after the filtering at the current model. The IMM uses (42) and (43) to merge the different initial values of those paths entering node $j$, $\hat{Z}^{(i)}_{k-1|k-1}$ and $\Sigma^{(i)}_{k-1|k-1}$, to a single initial point, $\bar{Z}^{(j)}_{k-1|k-1}$ and $\bar{\Sigma}^{(j)}_{k-1|k-1}$, according to a posterior probabilities before the filtering begins at the current model. Both the GPB2 and IMM force all those paths to start off from the same point (have identical initial values) for their expansion at next time $k+1$.

With the incorporation of the GPB2 and IMM, two dynamic-programming based decoding strategies have been developed. The one-pass dynamic programming decoding algorithm with the incorporation of the IMM merging strategy is described in Fig. 2 in its completion. The algorithm with GPB2 merging can be similarly described and is omitted here. In the remaining of this paper, we denote these two algorithms as GPB2-decoding (GPB2-D) and IMM-decoding (IMM-D), respectively.

[6] $l(\psi_K|O_1^K)$ here is equivalent to $P_K(\psi_K)$ in Section III.

## V. Experiments on Conversational Speech Recognition

In the experiments on conversational speech recognition designed to evaluate the decoding strategies presented in this paper, we choose one male speaker's data (speaker ID: 1028) extracted from **"train-ws97-a"** training set of the Switchboard corpus for training the switching state-space model for the speech dynamics. The training data consist of 966 utterances and about half an hour data. One single MLP was trained for the entire recognizer, common for all phones [5], [12]. One HMM system was also trained on the half an hour data, we call it **HMM-baseline** system [5].

N-best re-scoring paradigm was used for the evaluation of the new decoding approaches. The test set is chosen to be the male side of **"test-ws97-dev-l"** Switchboard test set (a total of 23 male speakers, 24 conversation sides, 1243 utterances and 50 min of speech). [5] One hundred hypotheses for each utterance in the test set have been generated by a state-of-the-art HMM system which was developed for the Workshop'97[7]. The phone segmentations generated by the HMM system are not consistent with our new model's dynamic regimes. In our earlier work [5], [12], the computation of the acoustic likelihoods for each hypothesis transcription in the 100-best list was carried out using the dynamic regimes fixed suboptimally from the phone boundaries provided by the HMM system. In this work the computation of the acoustic likelihoods is carried out using the dynamic regimes optimized by the decoding strategies. For decoding, each hypothesis becomes a simple lattice, where each phone is only allowed to transit to itself and to the next phone.

In order to focus on the acoustic modeling issue, we ignore the language model scores in our experiments. However, any improvement due to language models should be equally applicable to the new model.

We have tested the GPB2-decoding and IMM-decoding algorithms on simulated data. The results show that they both are generally capable of recovering the hidden switching time points used to artificially generate the observation data. Also, consistent with the theory, the GPB2-decoding algorithm is slightly more accurate but significantly slower than the IMM-decoding algorithm. Given the correct implementation of these algorithms as confirmed in these simulation experiments, we apply them to real speech data and carry out speech recognition experiments.

### A. Comparison Experiments

We first carry out a set of comparison experiments on a small set of the Switchboard test data, which consist of 240 utterances extracted randomly out of the male side of **"test-ws97-dev-l"** test set. We compare the two decoding algorithms described in this paper and the path-stack decoding (PS-D) algorithm described in detail in [24]. The PS-D algorithm uses a path deletion strategy, distinct from the path-merging one, to reduce the search space.

Table I lists the re-scoring word error rates, with (Ref $+100$) and without (100-best) adding the reference transcriptions to the 100-best lists. The "Baseline" recognizer computes the likelihoods of utterances using the dynamic regimes fixed from the phone boundaries provided by the HMM system. Recognizers labeled by "PS-D," "GPB2-D," and "IMM-D" represent the systems using the dynamic regimes optimized by the PS-D, GPB2-D, and IMM-D methods, respectively.

[7]See http://www.clsp.jhu.edu/ws97/ws97_general.html

TABLE I
Word Error Rates (%) for Four Recognizers With Different Methods of Likelihood Computation Using the N-Best Re-Scoring Paradigm

| Recognizers | Ref+100 | 100-best |
|---|---|---|
| Baseline | 53.2 | 60.6 |
| PS-D | 51.1 | 60.2 |
| GPB2-D | 50.1 | 60.1 |
| IMM-D | 49.5 | 59.7 |

TABLE II
Evaluation Results of IMM-D on the Whole Test Set

| Recognizers | Ref+100 | 100-best |
|---|---|---|
| Baseline | 55.6 | 58.3 |
| IMM-D | 51.5 | 57.1 |
| HMM-baseline | 56.1 | 58.9 |

From the recognition accuracy results shown on Table I, we observe that all three PS-D, GPB2-D, and IMM-D recognizers outperform the baseline recognizer, specially for the "Ref $+100$" case. The three decoding algorithms give comparable recognizer performance, with the IMM-D being a slightly better one. The reason that the GPB2-D has not produced better results than the IMM-D is probably due to the high level of noise in the real system, which makes the GPB2 approach lose estimation accuracy.

We summarize here a comparison among the GPB2, IMM, and PS decoding methods in terms of computation requirement. The IMM-D is much more efficient than the GPB2-D and PS-D. At each time step, the PS-D method runs a total of $S * M * M$ parallel EKFs ($S$ is the path-stack size, which is bigger than one) [24], while the GPB2-D method runs $M * M$ EKFs and the IMM-D method runs only $M$ EKFs.

Based on the most desirable accuracy and the computational requirement achieved by the IMM-D approach from the above results, we choose to use the IMM-D approach in a larger-scale speech recognition experiment, described below, using the entire set of the test data.

### B. Evaluation Experiments

The entire test set in the Switchboard corpus, consisting of 1243 utterances, is used in the evaluation experiments. The percent word error rate results are shown in Table II. The "Baseline" in Table II is a system constructed by using the nonlinear state-space model with the state switching times suboptimally fixed based on the HMM output. Compared with the "Baseline" system, the "IMM-D" algorithm gives about 4% absolute WER reduction for the "Ref $+100$" case and 1.2% for the "100 best" case.

The re-scoring results of the baseline HMM system under identical conditions are also given in the last row of the table. With the use of the "IMM-D" decoding algorithm, the nonlinear state-space model outperforms the HMM baseline system.

## VI. Summary and Discussion

In this paper, we report our continuing efforts in the development of a specialized, nonlinear state-space model for the VTR speech dynamics. The purpose of this development is targeted at its use in conversational speech recognition. The specific contribution of this work is the establishment of solid path merging strategies for recognizer decoding with the optimized dynamic regimes, one associated with each discrete state, in the speech model. Two specific dynamic-programming based algorithms using the merging strategies (GPB2-D and IMM-D) are presented. They both have successfully overcome the formidable exponential growth in the original search space. The new strategies are drastically different from the decoding techniques reported in [24] and in [12]. In [12], the phone boundaries in each of the N-best hypotheses need to be known in advance (based on the HMM segments in practical implementation), so decoding can be reduced to simple computation of the segment-bound model likelihoods. The work in [24] overcomes such a limitation via the use of aggressive path pruning in dynamic programming search. The strategies presented in this paper use path merging to cut down computation, without doing explicit path pruning. Speech recognition experiments using a subset of the Switchboard corpus have demonstrated that the use of the new decoding strategies can effectively reduce the recognizer's word error rate over the above two baseline recognizers under identical test conditions.

Both the GPB2-D and IMM-D decoding algorithms use merging strategies to limit the theoretically exponential growth of the search space. The merging formulas for both the GPB2 and IMM methods are theoretically motivated, being derived from Bayes' rule. The consequence of the merging is direct applicability of the dynamic programming principle to recognizer decoding, which would otherwise be impossible due to the essential VTR continuity constraint imposed across speech units in the construction of the speech model.

Taking into account the special temporal-flow properties in speech production and special requirements in speech recognition, we have developed two innovative ways of applying the switching state-space model, both distinct from the traditional approaches developed in the control-engineering and econometrics applications. First, in the traditional applications, no constraints were imposed on the model switching process; i.e., the model parameters can arbitrarily switch from one discrete state to another with nonzero probability at any time. For example, in the models of [17] and [22], the modeled economy may relatively freely switch from recession to booming and vice versa. In the model of [32], the measurement of the sensor may be the trajectory of any one of the multiple targets at each time. In contrast, for our applications of the dynamic model to speech production and recognition, specific structural constraints reflecting dynamic speech properties are imposed. Second, as a direct consequence of the left-to-right structural constraint for the switching process, the search for the "global" optimal discrete-state sequence (i.e., the switching history) based on the observation data becomes significantly more complex than the case without such a structural constraint. In the traditional applications [6], [17], [18], [22], [31], [32], [35], exclusive attention was paid to obtain the optimal "local" state estimation under the switching condition. In contrast, for the speech recognition application as our subject of study, the interest is in how the optimal switching process develops or how to find the global optimal discrete-state transition path (i.e., the optimal phone or word sequence).

Since currently the HMM is the most popular approach to speech recognition, it is important to compare the new state-space dynamic modeling approach with the HMM approach in terms of recognition time and number of parameters. The former is related to computation cost, which would be much greater (four orders of magnitude) for the state-space model than HMM if no approximation is made. After the most efficient IMM approximation is introduced, the difference can be reduced to roughly one order of magnitude. As for the parameter size or the memory cost, the new state-space model can be three orders of magnitude more economical compared with the context-dependent HMMs, and is in the same order of magnitude compared with the context-independent HMMs.

With the efficient decoding algorithms developed and described in this paper, it becomes possible to move the evaluation of the dynamic model from N-best list re-scoring to lattice re-scoring and possibly to full decoding in speech recognition. The algorithms can also be incorporated into the model training process to improve the training; that is, after each EM iteration, use the decoding algorithms to re-align the dynamic regimes before parameter estimation takes place. Another future direction for the development of the current speech model and recognizer is to adopt a full Bayesian approach to the model construction where both the "rate" and "target" parameters will be made randomly distributed (with continuous distributions) and are adaptive by on-line algorithms. While the model construction as presented in this paper allows these model parameters to be "random" with a discrete distribution, each mode in such a distribution is associated with a separate speech class (phone). For a fixed speech class, no parameter randomness has been permitted. This has limited the ability of the current model for embracing speaker variation in the VTR "target" and "hyper"-articulated speaking rate variation in the system dynamic "rate." By incorporating continuous distributions for model parameters within each phonetic class, such variations will be naturally accounted for, and hence more significant recognizer performance improvement will be expected.

## Appendix

*Derivation of (17) and (18):* By total probability theorem

$$p(Z(k)|O_1^k) = \sum_{n\in\Psi_k} P_k(n)p(Z(k)|\psi_k(n), O_1^k). \tag{48}$$

Hence,

$$\begin{aligned}\hat{Z}_{k|k} =& E\left[Z(k)|(O_1^k)\right] \\ =& \sum_{n\in\Psi_k} P_k(n)E\left[Z(k)|\psi_k(n), O_1^k\right] \\ =& \sum_{n\in\Psi_k} P_k(n)\hat{Z}_{n,k|k}\end{aligned} \tag{49}$$

and

$$\begin{aligned}\Sigma_{k|k} =& E\left[\left(Z(k)-\hat{Z}_{k|k}\right)\left(Z(k)-\hat{Z}_{k|k}\right)' |O_1^k\right]\\ =& \sum_{n\in\Psi_k} P_k(n) E\\ &\times\left[\left(Z(k)-\hat{Z}_{k|k}\right)\left(Z(k)-\hat{Z}_{k|k}\right)' |\psi_k(n), O_1^k\right]\\ =& \sum_{n\in\Psi_k} P_k(n) E\\ &\times\left[\left(\left(Z(k)-\hat{Z}_{n,k|k}\right)+\left(\hat{Z}_{n,k|k}-\hat{Z}_{k|k}\right)\right)\right.\\ &\cdot\left(\left(Z(k)-\hat{Z}_{n,k|k}\right)\right.\\ &\left.\left.+\left(\hat{Z}_{n,k|k}-\hat{Z}_{k|k}\right)\right)' |\psi_k(n), O_1^k\right]\\ =& \sum_{n\in\Psi_k} P_k(n)\\ &\times\left\{\Sigma_{n,k|k}+\left[\hat{Z}_{n,k|k}-\hat{Z}_{k|k}\right]\left[\hat{Z}_{n,k|k}-\hat{Z}_{k|k}\right]'\right\}.\end{aligned} \tag{50}$$

*Derivation of (19):* By Bayes' rule (see (51) at the bottom of the page).

*Derivation of (29) and (30) and (31):* By Bayes' rule

$$\begin{aligned}p\left(s_{k-1}=i|s_k=j,O_1^k\right)&=\frac{p\left(O(k),s_{k-1}=i|s_k=j,O_1^{k-1}\right)}{p\left(O(k)|s_k=j,O_1^{k-1}\right)}\\ &=\frac{p\left(O(k),s_{k-1}=i,s_k=j|O_1^{k-1}\right)}{p\left(O(k),s_k=j|O_1^{k-1}\right)}\\ &=\frac{p\left(O(k),s_{k-1}=i,s_k=j|O_1^{k-1}\right)}{\sum_{i=1}^{M} p\left(O(k),s_{k-1}=i,s_k=j|O_1^{k-1}\right)}\end{aligned} \tag{52}$$

$$\begin{aligned}p\left(s_k=j|O_1^k\right)&=\sum_{i=1}^{M} p\left(s_{k-1}=i,s_k=j|O_1^k\right)\\ &=\frac{\sum_{i=1}^{M} p\left(O(k),s_{k-1}=i,s_k=j|O_1^{k-1}\right)}{p\left(O(k)|O_1^{k-1}\right)}\\ &=\frac{\sum_{i=1}^{M} p\left(O(k),s_{k-1}=i,s_k=j|O_1^{k-1}\right)}{\sum_{j=1}^{M}\sum_{i=1}^{M} p\left(O(k),s_{k-1}=i,s_k=j|O_1^{k-1}\right)}\end{aligned} \tag{53}$$

and

$$\begin{aligned}&p\left(O(k),s_{k-1}=i,s_k=j|O_1^{k-1}\right)\\ &\quad=p\left(O(k),s_k=j|s_{k-1}=i,O_1^{k-1}\right)\\ &\qquad\times p\left(s_{k-1}=i|O_1^{k-1}\right)\\ &\quad=p\left(O(k)|s_{k-1}=i,s_k=j,O_1^{k-1}\right)\\ &\qquad\times p\left(s_k=j|s_{k-1}=i,O_1^{k-1}\right)p\left(s_{k-1}=i|O_1^{k-1}\right)\\ &\quad=p\left(O(k)|s_{k-1}=i,s_k=j,O_1^{k-1}\right)\\ &\qquad\times p\left(s_k=j|s_{k-1}=i\right)p\left(s_{k-1}=i|O_1^{k-1}\right).\end{aligned} \tag{54}$$

*Derivation of (47):* By total probability theorem

$$\begin{aligned}&p\left(Z(k)|s_k=j,O_1^{k-1}\right)\\ &\quad=\sum_{i=1}^{M} p\left(Z(k)|s_k=j,s_{k-1}=i,O_1^{k-1}\right)\\ &\qquad\times P\left(s_{k-1}=i|s_k=j,O_1^{k-1}\right)\\ &\quad=\sum_{i=1}^{M}\left[\int p\left(Z(k)|Z(k-1),s_k=j,s_{k-1}=i,O_1^{k-1}\right)\right.\\ &\qquad\left.\times p(Z(k-1)|s_k=j,s_{k-1}=i,O_1^{k-1}dZ(k-1)\right]\\ &\qquad\times P\left(s_{k-1}=i|s_k=j,O_1^{k-1}\right)\\ &\quad=\int p\left(Z(k)|Z(k-1),s_k=j,O_1^{k-1}\right)\\ &\qquad\times\left[\sum_{i=1}^{M} p\left(Z(k-1)|s_{k-1}=i,O_1^{k-1}\right)\right.\\ &\qquad\left.\times P\left(s_{k-1}=i|s_k=j,O_1^{k-1}\right)\right]dZ(k-1).\end{aligned} \tag{55}$$

$$\begin{aligned}P_k(n) =& P(\psi_k(n)|O_1^k)\\ =& \frac{P(O(k),s_{n,k}=i_{n,k},\psi_{k-1}(m)|O_1^{k-1})}{P(O(k)|O_1^{k-1})}\\ =& \frac{P(O(k),s_{n,k}=i_{n,k}|\psi_{k-1}(m),O_1^{k-1})P(\psi_{k-1}(m)|O_1^{k-1})}{P(O(k)|O_1^{k-1})}\\ =& \frac{P(O(k)|s_{n,k}=i_{n,k},\psi_{k-1}(m),O_1^{k-1})P(s_{n,k}=i_{n,k}|\psi_{k-1}(m),O_1^{k-1})}{P(O(k)|O_1^{k-1})}P_{k-1}(m)\\ =& \frac{P(O(k)|\psi_k(n),O_1^{k-1})P(s_{n,k}=i_{n,k}|\psi_{k-1}(m),O_1^{k-1})}{P(O(k)|O_1^{k-1})}P_{k-1}(m)\end{aligned} \tag{51}$$

## REFERENCES


[1] B. Anderson and J. Moore, *Optimal Filtering*. Englewood Cliffs, NJ: Prentice-Hall, 1979.

[2] R. Bakis, "Coarticulation modeling with continuous-state HMM's," in *Proc. IEEE Workshop Automatic Speech Recognition*, Arden House, NY, 1991, pp. 20–21.

[3] C. Blackburn and S. Young, "Toward improved speech recognition using a speech production model," in *Proc. Eurospeech*, vol. 2, 1995, pp. 1623–1626.

[4] H. Bourlard, H. Hermansky, and N. Morgan, "Toward increasing speech recognition error rates," *Speech Commun.*, vol. 18, pp. 205–231, 1996.

[5] J. Bridle, L. Deng, J. Picone, H. Richards, J. Ma, T. Kamm, M. Schuster, S. Pike, and R. Reagan, "An investigation of segmental hidden dynamic models of speech coarticulation for automatic speech recognition," in *Final Report for the 1998 Workshop on Language Engineering*, 1998, pp. 1–61.

[6] C. B. Chang and M. Athans, "State estimation for discrete systems with switching parameters," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-14, no. 3, pp. 418–423, 1978.

[7] J. Cohen, "The summers of our discontent," in *Proc. Addendum ICSLP*, 1996, pp. S9–10.

[8] L. Deng, "A generalized hidden Markov model with state-conditioned trend functions of time for the speech signal," *Signal Process.*, vol. 27, pp. 65–78, 1992.

[9] L. Deng and M. Aksmanovic, "Speaker-independent phonetic classification using hidden Markov models with state-conditioned mixtures of trend functions," *IEEE Trans. Speech Audio Processing*, vol. 5, pp. 319–324, July 1997.

[10] L. Deng, "A dynamic, feature-based approach to the interface between phonology and phonetics for speech modeling and recognition," *Speech Commun.*, vol. 24, no. 4, pp. 299–323, 1998.

[11] ——, "Computational models for speech production," in *Computational Models of Speech Pattern Processing (NATO ASI Series)*. New York: Springer, 1999, pp. 214–224.

[12] L. Deng and J. Z. Ma, "Spontaneous speech recognition using a statistical coarticulatory model for the vocal-tract-resonance dynamics," *J. Amer. Soc. Acoust.*, vol. 108, no. 6, pp. 3036–3048, Dec. 2000.

[13] V. Digalakis, M. Ostendorf, and J. Rohlicek, "Fast algorithm for phone classification and recognition using segment-based models," *IEEE Trans. Speech Audio Processing*, pp. 2885–2896, Dec. 1992.

[14] V. Digalakis, J. Rohlicek, and M. Ostendorf, "ML estimation of a stochastic linear system with the EM algorithm and its application to speech recognition," *IEEE Trans. Speech Audio Processing*, vol. 1, pp. 431–442, 1993.

[15] Y. Gao, R. Bakis, J. Huang, and B. Xing, "Multistage coarticulation mdoel combining articulatory, formant and cepstral features," in *Proc. ICSLP*, Bejing, China, 2000, pp. 25–28.

[16] Z. Ghahramani and G. Hinton, "Variational learning for switching state-space model," *Neural Comput.*, vol. 12, pp. 831–864, 2000.

[17] J. D. Hamilton, "A new approach to the economic analysis of nonstationary time series and the business cycle," *Econometrica*, vol. 57, no. 2, pp. 357–384, 1989.

[18] P. J. Harrison and C. F. Stevens, "Bayesian forecasting," *J. R. Statist. Soc. B*, vol. 38, pp. 205–247, 1976.

[19] W. Holmes and M. Russell, "Probabilistic-trajectory segmental HMM's," *Comput. Speech Lang.*, vol. 13, pp. 3–37, 1999.

[20] J. Holmes, W. Holmes, and P. Garner, "Using formant frequencies in speech recognition," in *Proc. Eurospeech*, vol. 4, 1997, pp. 2083–2086.

[21] F. Jelinek, *Statistical Methods for Speech Recognition*. Cambridge, MA: MIT Press, 1998.

[22] C. J. Kim, "Dynamic linear models with Markov-switching," *J. Econometrics*, vol. 60, pp. 1–22, 1994.

[23] K. Kitagawa, "Non-Gaussian state-space modeling of nonstationary time series," *J. Amer. Statist. Assoc.*, vol. 82, pp. 1032–1041, 1987.

[24] J. Ma and L. Deng, "A path-stack algorithm for optimizing dynamic regimes in a statistical hidden dynamic model of speech," *Comput., Speech, Language*, vol. 14, pp. 101–114, 2000.

[25] J. M. Mendel, *Lessons in Estimation Theory for Signal Processing Communication and Control*. Englewood Cliffs, NJ: Prentice-Hall, 1995.

[26] R. Moore, "Speech pattern processing," in *Computational Models of Speech Pattern Processing (NATO ASI)*: Springer, 1999, pp. 1–9.

[27] M. Ostendorf, V. Digalakis, and O. Kimball, "From HMM's to segment models: A unified view of stochastic modeling for speech recognition," *IEEE Trans. Speech Audio Processing*, vol. 4, pp. 360–378, 1996.

[28] J. Picone, S. Pike, R. Reagan, T. Kamm, J. Bridle, L. Deng, J. Ma, H. Richards, and M. Schuster, "Initial evaluation of hidden dynamic models on conversational speech," in *Proc. ICASSP*, Mar. 1999, pp. 109–112.

[29] H. Richards and J. Bridle, "The HDM: A segmental hidden dynamic model of coarticulation," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Mar. 1999, pp. 9–12.

[30] Y. Bar-Shalom and T. E. Fortmann, *Tracking and Data Association*. New York: Academic, 1988.

[31] Y. Bar-Shalom and X. R. Li, *Estimation and Tracking*. Boston, MA: Artech House, 1993.

[32] R. H. Shumway and D. S. Stoffer, "Dynamic linear models with switching," *J. Amer. Statist. Assoc.*, vol. 86, no. 415, p. .

[33] M. Siu, R. Iyer, H. Gish, and C. Quillen, "Parametric trajectory mixtures for LVCSR ," in *Proc. ICSLP*, Sydney, Australia, 1998, pp. 3269–3272.

[34] H. Tanizaki, *Nonlinear Filters* , 2nd ed. New York: Springer-Verlag , 1996.

[35] B. Anderson and J. Moore, "Survey of Bayesian and non-Bayesian testing of model stability in econometrics," in *Bayesian Analysis of Time Series and Dynamic Linear Models* , J. C. Spall, Ed. New York: Marcel-Dekker, 1988.

[36] K. Watanabe, *Adapative Estimation and Control—Partitioning Approach*. Englewood Cliffs, NJ: Prentice-Hall, 1991.


**Jeff Z. Ma** received the B.Sc. degree in electrical engineering from Xi'an Jiaotong University, China in 1989, the M.Sci. degree in pattern recognition from Chinese Academy of Sciences, China, in 1992, and the Ph.D degree in electrical and computer engineering from University of Waterloo, Waterloo, ON, Canada in 2000.

He was Research Assistant in the Department of Computer Science, University of Hong Kong, from 1993 to 1995, and Research Assistant in the Department of Electrical and Computer Engineering, University of Waterloo, 1996 to 2000. In 2000, he joined BBN Technologies, Cambridge, MA. He has been working on conversational speech recognition on different languages (English, Mandarin, Arabic), speech recognition over IP channels, Mandarin broadcast news audio indexing, and topic identification. His current research interests include conversational speech recognition, topic classification, discriminative training, and dynamic models for speech recognition.

**Li Deng** (S'83–M'86–SM'91) received the B.S. degree from University of Science and Technology of China in 1982, the M.S. degree from the University of Wisconsin—Madison in 1984, and the Ph.D. degree from the University of Wisconsin—Madison in 1986.

He worked on large vocabulary automatic speech recognition at INRS-Telecommunications in Montreal, Montreal, QC, Canada, from 1986 to 1989. In 1989, he joined the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada as Assistant Professor; he became Full Professor in 1996. From 1992 to 1993, he conducted sabbatical research at Laboratory for Computer Science, Massachusetts Institute of Technology, Cambridge, and from 1997 to 1998, at ATR Interpreting Telecommunications Research Laboratories, Kyoto, Japan. In 1999, he joined Microsoft Research, Redmond, WA, as Senior Researcher, and is currently a Principal Investigator in the DARPA-EARS program and affiliate Professor of Electrical Engineering at University of Washington. His research interests include acoustic-phonetic modeling of speech, speech and speaker recognition, speech synthesis and enhancement, speech production and perception, auditory speech processing, noise robust speech processing, statistical methods and machine learning, nonlinear signal processing, spoken language systems, multimedia signal processing, and multimodal human–computer interaction. In these areas, he has published over 200 technical papers and book chapters, and has given keynote, tutorial, and other invited lectures. He recently completed the book *Speech Processing—A Dynamic and Optimization-Oriented Approach* (New York: Marcel Dekker, 2003).

Dr. Deng served on Education Committee and Speech Processing Technical Committee of the IEEE Signal Processing Society during 1996-2000, and is currently serving as Associate Editor for the IEEE TRANSACTIONS ON SPEECH AND AUDIO PROCESSING. He is a Technical Chair of the 2004 International Conference on Acoustics, Speech, and Signal Processing (ICASSP).