THE HDM: A SEGMENTAL HIDDEN DYNAMIC MODEL OF COARTICULATION

Hywel B. Richards and John S. Bridle

Dragon Systems UK, Millbank, Stoke Road, Cheltenham, GL52 4RW, UK.
hywel@dragon.co.uk

ABSTRACT

This paper introduces a new approach to acoustic-phonetic modelling, the Hidden Dynamic Model (HDM), which explicitly accounts for the coarticulation and transitions between neighbouring phones. Inspired by the fact that speech is really produced by an underlying dynamic system, the HDM consists of a single vector target per phone in a hidden dynamic space in which speech trajectories are produced by a simple dynamic system. The hidden space is mapped to the surface acoustic representation via a non-linear mapping in the form of a multilayer perceptron (MLP). Algorithms are presented for training of all the parameters (target vectors and MLP weights) from segmented and labelled acoustic observations alone, with no special initialisation. The model captures the dynamic structure of speech, and appears to aid a speech recognition task based on the SwitchBoard corpus.

1. INTRODUCTION

Much of the complexity and indirectness of the relationship between the acoustic patterns of speech and the linguistic structures that they represent is caused by context-sensitivity [1].

Conventional large vocabulary continuous speech recognition systems model speech patterns as a sequence of stationary segments (albeit with differential features). Various effects, including phonological variation and hard coarticulation, are dealt with by dividing the contexts of each phoneme into equivalence classes (using context decision trees) and modelling these contextual variants separately (using mixtures of Gaussian components). While this is a reasonable procedure for some types of phonological variation, it is an extravagant approach to coarticulation, and ignores some well-known properties of real human speech, with the result that very large amounts of training material are required if the system is to perform well with a large vocabulary and a variety of speakers and speaking styles.

We are interested in a principled approach to the coarticulation problem in acoustic-phonetic modelling that offers to provide compact, realistic models that generalize well. It is a synthesis-based method in which each phonetic segment has a (vector-valued) target characteristic of the type of segment. A dynamic process 'smooths' the sequence of targets, to produce a trajectory analogous to formant frequency tracks or articulator positions. The observed acoustic pattern is produced through a non-linear mapping.

There have been several attempts to find useful alternatives to the frame-by-frame finite-state HMM systems, so that whole phonetic segments are treated specially, and so that coarticulation is a natural consequence of the model. One of the most general was by Bakis [2]. Most attempts have used a linear mapping between the space in which the dynamics happens and the acoustic observations [3]. Our approach is similar to that of Blackburn [4] in that we use an MLP, but we use a single MLP and our hidden dynamic system is much simpler yet quite powerful.

2. THE HDM AS A SPEECH SYNTHESISER

The Hidden Dynamic Model describes the way in which an acoustic pattern is produced from a sequence of phones with given durations. The structure of the HDM is shown in Figure 1.

For each phone class, there is a single target vector which defines a point in the hidden dynamic space. For each phone segment in the sequence, the respective target applies for the duration of that segment, resulting in the target sequence, t_j, shown in Figure 1. This is typically multidimensional, but a single dimension is shown here for clarity.

Note that the symbols we call 'phones' here are supposed to correspond to acoustic segments with one target – diphthongs and some allophones need more than one phone.

This target sequence is smoothed to produce a trajectory in hidden dynamic space, x_j. The filter used for this smoothing is a second-order symmetrical (forward-backward) low-pass filter, whose single time-constant parameter, p_j, is also determined by the phone class (see Appendix). In the general multidimensional case, there is different time-constant for each dimension of the hidden dynamic space (the motivation for this will be described later).

The hidden dynamic trajectory is mapped to the surface acoustic form, y_j, by a non-linear mapping, here a multi-layer perceptron (MLP). This mapping defines the hidden dynamic space, and a single MLP is used for all phones. This mapping can be considered analogous to the mapping between vocal tract shapes and speech sounds, although we intend it to be learned only from the acoustic data (as described in the next section), and not restrict it to any predetermined form. The criterion is to model the structure of the acoustic speech pattern.

3. TRAINING

This section describes how all of the parameters of the HDM can be learned from segmented and labelled acoustic training data.

In training, the HDM is again used to synthesize an acoustic pattern, y_j from a sequence of phone symbols and timings (Figure 1). Now, however, we calculate how much the synthetic pattern differs from the acoustic training data, in the simplest case, using a Euclidean distance, $E = \sum_j |\underline{y}_j - \underline{z}_j|^2$, where j ranges over the complete training corpus.

In order to improve the acoustic pattern synthesis, we want to change the HDM parameters so that this error, E, is reduced. One way to achieve this is to obtain the derivatives of the error with respect to each parameter of the model, and then carry out some form of gradient descent on E.

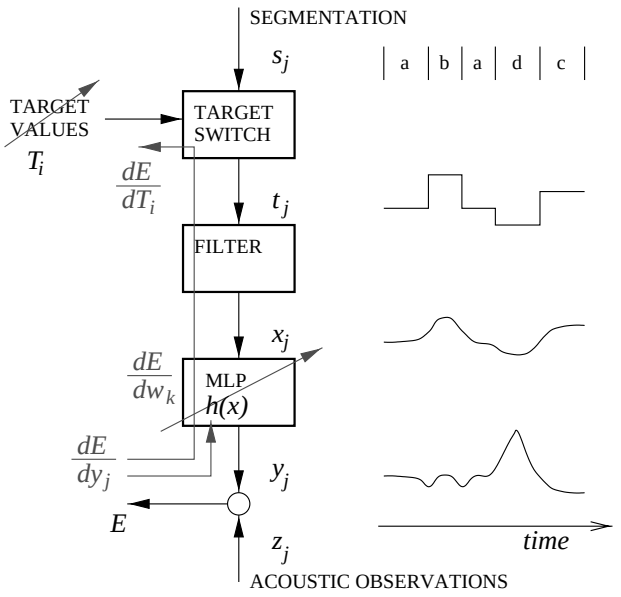

Figure 1: Calculating an acoustic error using the Hidden Dynamic Model. Also shown is how derivatives of this error are backpropagated through to the MLP weights and target vectors.

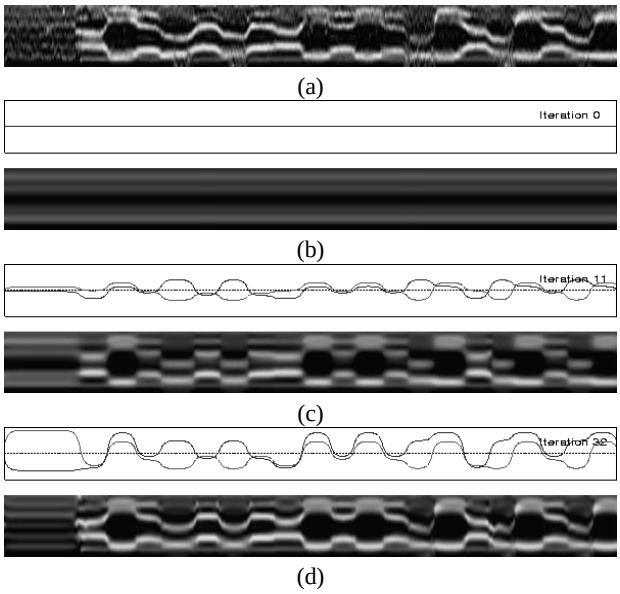

Figure 2: The training acoustic pattern $\underline{z}_j$ (a), and the hidden dynamic trajectory $\underline{x}_j$ and synthetic speech $\underline{y}_j$ at different points in the training procedure: (b) at initialisation, (c) after 11 iterations, and (d) after 32 iterations.

Derivatives of E can be backpropagated through to the MLP weights in exactly the same way this is usually done when training an MLP from input/output pairs, first obtaining derivatives of E with respect to the MLP outputs. This allows us to obtain $\frac{\partial E}{\partial w_k}$, for all of the MLP weights w_k [5].

If the derivatives of the error are backpropagated right back to the MLP inputs to give $\frac{\partial E}{\partial \underline{x}_j}$, it is possible to backpropagate the error derivatives further, through the smoothing function to the phone target values and time constants to obtain $\frac{\partial E}{\partial \underline{T}_i}$ and $\frac{\partial E}{\partial \underline{P}_i}$ (see Appendix).

Once the derivatives of E with respect to all of the HDM parameters have been obtained, the parameters can be optimized using gradient descent. All that remains to decide is what initialisation, if any, is required.

Figure 2 shows hidden dynamic trajectories and synthetic acoustic patterns produced at various stages in such a gradient descent learning procedure.

For the purposes of simplicity, the training data here is an 11 second connected vowel utterance, of which roughly two-thirds is shown. The acoustic patterns shown in Figure 2 are spectrograms derived from the 12th order MFCC representation used for $\underline{y}_j$ and $\underline{z}_j$. This training data, together with a segmentation and labelling is presented to the HDM training algorithm.

The HDM target parameters consist of six two-dimensional hidden space targets for: (a) the five phone classes in the training data, and (b) an additional target for silence. The two-dimensional nature of the hidden space permits an easy visualisation of the hidden dynamic trajectory (which will be shown later). All of these targets are initialized at zero (which results in the rather uninteresting zero-valued hidden 'trajectory' shown in Figure 2(b)). The MLP here has 40 hidden units in one hidden layer, and is initialized with small random weight values. The time-constants are fixed here to $\underline{P}_i = [3, 3]$ frames for all phone classes, i.

After 31 iterations of conjugate gradient descent [5], the optimisation stops and the HDM is now capable of reproducing the training data with a reasonable accuracy (Figure 2(d)). Visual inspection of synthetic spectrograms produced by this HDM using other phone sequences reveal that it is also capable of producing plausible transitions for phone combinations not seen in the training data.

4. THE HIDDEN SPACE

Because all the time constant parameters are equal in the HDM described in the previous section, the hidden dynamic space can be an arbitrary linear transform of these $\underline{x}$ parameters to give the same results. In Figure 3(a), the $\underline{x}_j$ trajectories have been offset and scaled to show that they closely resemble the formant frequencies on a warped frequency scale.

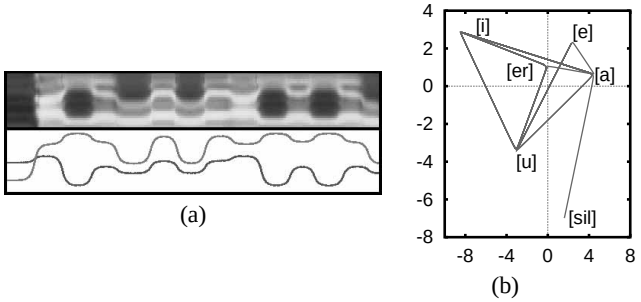

Figure 3: The synthetic spectrogram ((a) top) and the two hidden dynamic parameters ((a) bottom and (b)). In this case, the HDM has discovered the formant frequencies on a warped frequency scale.

Figure 4: Synthetic speech produced by an HDM with using a linear mapping instead of an MLP. The HDM here has been trained using the same material as in Figure 2.

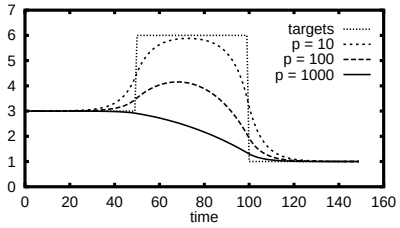

Figure 5: Hidden dynamic trajectories with time constants of 10, 100 and 1000 units for the centre segment.

The hidden dynamic trajectory is plotted in the original unscaled hidden space in Figure 3(b). The vertices of this trajectory correspond to the phone targets, and because the vowels have been produced quite slowly in this utterance, the trajectory approaches very close to the target for each vowel. It can be seen that the x-axis corresponds to F_1, while the y-axis corresponds to F_2.

Also of interest here is the way in which the hidden dynamic model encodes the silence 'phone' in the hidden representation, using a usually undefined region of formant space where F_2 drops below F_1.

5. THE IMPORTANCE OF BEING NON-LINEAR

Figure 4 shows synthetic speech produced from an HDM, which has been trained with the same training data as in Section 3, but uses a linear mapping in place of the MLP used in Section 3.

Using a linear mapping here is equivalent to using the simple transitions given by the dynamic system, but operating directly in the acoustic domain. The linear system is unable to reproduce convincing formant transitions like those seen in Figure 2(d). This demonstrates the importance of the non-linear mapping in modelling the dynamics of the speech pattern.

6. DYNAMICS

The Appendix gives details of the time-varying symmetrical smoothing that constitutes the dynamic part of the HDM. The time-constants can also be thought of as target importance weights, variances associated with the targets, or the strength of influence of the targets on the trajectory. Figure 5 shows that by varying the time-constant for a segment, the trajectory can either approach the the target closely, or almost ignore it.

We chose this form of smoothing so that it would be useful for some aspects of consonant gestures. For example, in the production of a bilabial consonant such as the voiced stop, /b/, the most important articulatory action is to make a closure at the lips. The shape of the remainder of the vocal tract, such as the tongue position, is determined mainly by the context in this case [6]. According to the 'critical articulator' theory [7], the production of other consonants is similar, with the consonant influencing usually only a local region of the vocal tract. This is in contrast to the way vowel sounds specify an overall shape to the vocal tract.

Although the hidden representation of the HDM is an abstract hidden space, which is derived only from the data used to train the HDM, the time constant parameters, P_i, at least give it the flexibility to synthesize the sort of trajectories that occur in articulatory space.

7. USE IN LVCSR

The simplest way to use an HDM for large vocabulary speech recognition is to rescore N-best lists. Given word-transcription hypotheses obtained from a conventional recognizer, a phone sequence and alignment can be obtained by using the same HMM speech models used in the recognizer. The synthetic speech pattern produced by the HDM for each phone sequence can be compared with the observed speech, giving N new distance scores.

These scores, when combined with appropriate language model scores, can be used to select the best transcription from the N-best list according to the HDM. In general, the HDM scores can be combined with those from the conventional system [8].

In a small experiment [9] with a portion of the Switchboard corpus we observed that in choosing between the best 5 (or the best 100) word sequences the HDM rescoring was not useful, performing close to chance. However, when the best 5 were 'enriched' (by adding the reference transcript), the HDM system showed that it contained information not in the original HMM system (word error rate dropped from 48% to 35%). The 30 minutes of training material was from a single male speaker, and the test material was 1241 utterances from 23 other men. A simple HMM system trained on exactly the same material did not reduce the WER in this way.

8. CONCLUSIONS AND FUTURE WORK

We think that the simple, flexible structure described here has potential for capturing important aspects of the relationship between phonetic labels and acoustic patterns, with potential applications in speech science, synthesis and recognition. Because it is so simple, there are also many possibilities for extending it.

Although the system has been presented here as a trainable synthesizer, with a simple error measure used in re-scoring for ASR, it is not difficult in principle to pose it as a stochastic model with the squared error as a log likelihood of observations Gaussian-distributed about the mean for each frame, and to allow probabilistic variation in the segment targets.

Perhaps the most important area where our model needs to be developed concerns time-alignment. These days we expect an approach to ASR to include a method a dealing with unknown timescales, and solutions are usually based on Dynamic Programming, which relies on a finite state-space. DP is not applicable to search for HDMs, but there are several possibilities for sub-optimal search for alignment. One of the simplest is to find the timewarp that aligns the synthetic pattern and the natural pattern, then apply this timewarp to the segmentation [10].

Because of the economical parameterisation, there are some interesting possibilities for dealing with speaker differences. For instance we can hope that most of the variation in the sets of target vectors for different speakers will be included in a low-dimensional 'speaker space', in which speaker adaptation can be attempted [10].

We have some hope of extending the model to be compatible with modern phonological theories based on overlapping features, as proposed by Deng [11].

9. ACKNOWLEDGEMENTS

This material is based upon work supported by the National Science Foundation under Grant No. (#IIS-9732388), and was carried out at the 1998 Workshop on Language Engineering, Center for Language and Speech Processing, Johns Hopkins University. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the NSF or JHU.

APPENDIX

The forward-backward filter used in Section 2 is a simple form of Kalman smoother. It can be expressed for a single hidden space dimension as

$$\underline{x} = K(\underline{t}, \underline{p}, r) \tag{1}$$

where $\underline{x}$ is the hidden dynamic state sequence (x_j from $j = 0$ to $j = n-1$), $\underline{t}$ is the target sequence, $\underline{p}$ is the time-constant sequence, and r determines the scale of the time-constant values and is set to unity in this work.

This function essentially solves the set of n simultaneous equations acting at each frame j which balance the influence of the target t_j and the adjacent values of $\underline{x}$: x_{j+1} and x_{j-1}

$$x_j(p_j^{-1} + 2r^{-1}) = p_j^{-1}t_j + r^{-1}x_{j+1} + r^{-1}x_{j-1} \tag{2}$$

The forward-backward Kalman smoother equations which describe this function $K(\underline{t}, \underline{p}, r)$ are as follows:

'Forward prior' calculation:

$$\alpha_j^+ = \frac{p_j\alpha_{j-1}^+ + (\beta_{j-1}^+ + r)t_j}{p_j + (\beta_{j-1}^+ + r)}, \quad \beta_j^+ = \frac{p_j(\beta_{j-1}^+ + r)}{p_j + (\beta_{j-1}^+ + r)} \tag{3}$$

where $j = 1$ to $n-1$ and the recursion is initialized with

$$\alpha_0^+ = t_0, \quad \beta_0^+ = p_0 \tag{4}$$

α_j^+ and β_j^+ represent the prior mean and variance for the dynamic state given all of the targets up until frame j[1].

The 'backward prior' calculation to obtain α_j^- and β_j^- is simply the time-symmetric version of the forward one.

Forward and backward prior combination:

$$\begin{aligned} \alpha_j^{+-} &= \frac{\alpha_{j-1}^+(\beta_{j-1}^+ + r) + \alpha_{j+1}^-(\beta_{j+1}^- + r)}{(\beta_{j-1}^+ + r) + (\beta_{j+1}^- + r)}, \\ \beta_j^{+-} &= \frac{(\beta_{j-1}^+ + r)(\beta_{j+1}^- + r)}{(\beta_{j-1}^+ + r) + (\beta_{j+1}^- + r)} \end{aligned} \tag{5}$$

where $j = 1$ to $n-2$ and

$$\alpha_0^{+-} = \alpha_1^-, \qquad \beta_0^{+-} = \beta_1^- \tag{6}$$

$$\alpha_{n-1}^{+-} = \alpha_{n-2}^+, \quad \beta_{n-1}^{+-} = \beta_{n-2}^+ \tag{7}$$

α_j^{+-} and β_j^{+-} represent the prior mean and variance given the whole target sequence *except* the target at frame j.

Finally, the smoothed state sequence is derived by combining the forward-backward prior with the target 'observations'.

$$x_j = \frac{\alpha_j^{+-}p_j + t_j\beta_j^{+-}}{p_j + \beta_j^{+-}} \tag{8}$$

[1] Apologies to those accustomed to using α and β to represent the forward and backward probability distributions in conventional HMM calculations.

where x_j is the hidden dynamic state sequence at time j.

The Kalman smoother function, $K(\underline{t}, \underline{p}, r)$, is also used to back-propagate error derivatives to the HDM target, T_i, and time constant parameters, P_i, during training.

It can be shown that from Equation 2

$$\frac{\partial \underline{x}}{\partial T_i} = K(\frac{\partial \underline{t}}{\partial T_i}, \underline{p}, r) \tag{9}$$

The derivative of the error with respect to the target values can be obtained by the chain rule

$$\frac{\partial E}{\partial T_i} = \sum_{j=0}^{n-1} \frac{\partial E}{\partial x_j}\frac{\partial x_j}{\partial T_i} \tag{10}$$

Similarly, to obtain the derivatives of the error with respect to the time-constant parameters

$$\frac{\partial \underline{x}}{\partial P_i} = K(\underline{u}, \underline{p}, r) \tag{11}$$

where

$$u_j = \left(\frac{x_j - t_j}{p_j}\right)\frac{\partial t_j}{\partial T_i} \tag{12}$$

Again, the derivative of the error with respect to the P_i time-constant parameters can be obtained by the chain rule.

10. REFERENCES

[1] J. Clark and C. Yallop. *An introduction to phonetics and phonology*. Blackwell, Oxford and Cambridge, MA, 1990.

[2] R. Bakis. Coarticulation modeling with continuous-state HMMs. In *Proc. IEEE Workshop Automatic Speech Recognition*, pages 20–21, Arden House, New York, 1991.

[3] V. Digalakis, J. R. Rohlicek, and M. Ostendorf. A dynamical system approach to continuous speech recognition. In *Proc. ICASSP-91*, volume 1, pages 289–292, 1991.

[4] C. S. Blackburn and S. J. Young. Towards improved speech recognition using a speech production model. In *Proc. Eurospeech-95*, volume 3, pages 1623–1626, 1995.

[5] C. M. Bishop. *Neural networks for pattern recognition*. Oxford University Press, 1995.

[6] S. E. G. Öhman. Numerical model of coarticulation. *J. Acoust. Soc. Am.*, 41:310–320, 1967.

[7] G. Papcun, J. Hochberg, T. R. Thomas, F. Laroche, J. Zacks, and S. Levy. Inferring articulation and recognizing gestures from acoustics with a neural network trained on x-ray microbeam data. *J. Acoust. Soc. Am.*, 92(2):688–700, 1992.

[8] M. Ostendorf, A. Kannan, S. Austin, O. Kimball, S. Schwartz, and J.R. Rohlicek. Integration of diverse recognition methodologies through reevaluation of N-Best sentence hypotheses. In *Proc. DARPA Speech and Language Workshop*, pages 83–87, February 1991.

[9] J. Picone, S. Pike, R. Reagan, T. Kamm, J. Bridle, L. Deng, Z. Ma, H. Richards, and M. Schuster. Initial evaluation of hidden dynamic models on conversational speech. *submitted to ICASSP-99 (see also http://www.clsp.jhu.edu/ws98/)*, 1999.

[10] J.S. Bridle and M.R. Ralls. An approach to automatic speech recognition based on synthesis-by-rule. In F. Fallside and W. A. Woods, editors, *Computer speech processing*. Prentice Hall, 1985.

[11] L. Deng. A dynamic, feature-based approach to the interface between phonology and phonetics for speech modeling and recognition. *Speech Communication*, 24(4):299–323, 1998.