# NOISE ROBUST SPEECH RECOGNITION WITH A SWITCHING LINEAR DYNAMIC MODEL

*Jasha Droppo and Alex Acero*

Microsoft Research
Redmond, Washington 98052, USA

## ABSTRACT

Model based feature enhancement techniques are constructed from acoustic models for speech and noise, together with a model of how the speech and noise produce the noisy observations.

Most techniques incorporate either Gaussian mixture models (GMM) or hidden Markov models (HMM). This paper explores using a switching linear dynamic model (LDM) for the clean speech. The linear dynamics of the model capture the smooth time evolution of speech. The switching states of the model capture the piecewise stationary characteristics of speech.

However, incorporating a switching LDM causes the enhancement problem to become intractable. With a GMM or an HMM, the enhancement running time is proportional to the length of the utterance. The switching LDM causes the running time to become exponential in the length of the utterance. To overcome this drawback, the standard generalized pseudo-Bayesian technique is used to provide an approximate solution of the enhancement problem.

We present preliminary results demonstrating that, even with relatively small model sizes, substantial word error rate improvement can be achieved.

## 1. INTRODUCTION

Automatic speech recognition systems without explicit provisions for noise robustness degrade quickly in the presence of additive noise. As a consequence, how to best add noise robustness to such systems is an area of active research.

The system presented in this paper is one of a number of model based feature enhancement systems. Such systems include a model for speech, and often a model for noise as well, within the enhancement algorithm.

When the clean speech model is a Gaussian mixture model (GMM), each frame of data is enhanced independently. Without post-processing, this can result artifacts, such as sharp single frame transitions, that were not part of the original clean speech signal.

Choosing a hidden Markov model (HMM) for the clean speech model introduces some time dependencies in the enhancement process. Although, for any given state sequence, the enhancement process is the same as for a GMM, the state transition probabilities of the HMM tend to eliminate single frame errors in the output. State transitions can still produce edge artifacts, so some post-processing is still necessary.

This paper presents a framework for incorporating a switching linear dynamic model (LDM) for clean speech. Like the GMM or HMM that are normally used, the switching LDM maintains the concept that, as time progresses, the signal passes through several distinct states. In addition, the switching LDM enforces a continuous state transition in the feature space, conditioned on the state sequence.

The major obstacle to using the switching LDM for enhancement is the computational burden that it brings. If the clean speech model is a GMM or HMM, enhancement of a signal with length $T$ takes $O(T)$ time. However, the switching LDM produces an enhancement algorithm that takes $O(e^T)$ time. Even for short utterances, $T$ is on the order of several hundred frames, and the direct approach is infeasible. To overcome this obstacle, we show how the generalized pseudo-Bayesian technique [1] can be used to provide an approximate solution.

The work in [2] bears some similarity to the method described in this paper, with three important differences. First, the current paper explores using a switching LDM for speech, whereas the previous work considered it only for noise. Although there may be some noise types, such as babble, that may benefit from the power of the LDM, many stationary noise types would not.

Second, the approximate posteriors are quite different. The resampling approximation in [2] approximates the posterior as a set of discrete points in the feature space. The current paper represents the posterior as a sum of Gaussian components.

It also common to separate noise tracking and enhancement, as is done in [2]. We believe that joint noise and speech tracking should yield a better enhancement result, and that is the method followed by this paper.

We present preliminary results demonstrating that, even with relatively small model sizes, substantial word error rate improvement can be achieved. Section 2 describes the switching LDM used to model the speech and noise. Section 3 briefly describes the observation model that we use to unify the speech and noise models. Section 4 presents the method we use to approximate the posterior distribution of speech and noise, after the noisy observations are incorporated. Section 5 analyzes the effectiveness of different combinations of parameters on improving digit accuracy under the current test.

## 2. SYSTEM EQUATIONS

The first step in building the noise removal system is to define a set of system equations that describe the clean speech process.

### 2.1. Linear Dynamic Model

A standard LDM obeys the system equation,

$$x_t = Ax_{t-1} + b + v_t.$$

Here, $A$ and $b$ describe how the process evolves over time, and $v_t$ is a zero-mean Gaussian noise source which drives the system. LDM are time-invariant, and are useful in describing signals such as colored stationary Gaussian noise.

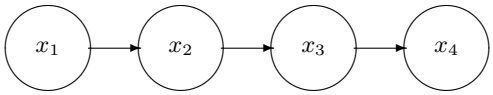


**Fig. 1**. Graphical representation of the LDM used for noise in this paper.

This type of system is typically presented as either the graphical model of Figure 1, or as the equations

$$p(x_t|x_{t-1}) = N(x_t; Ax_{t-1} + b, C)$$

$$p(x_1^T) = p(x_1) \prod_{t=2}^{T} p(x_t|x_{t-1})$$

### 2.2. Switching LDM

In a switching LDM, the $A$ and $b$ are dependent on a hidden variable at each time $t$.

$$x_t = A_{s_t} x_{t-1} + b_{s_t} + v_t.$$

Every unique state sequence $s_1^T$ describes a non-stationary LDM. As a result, it is appropriate for describing a number of time-varying systems, including the evolution of speech and noise features over time.

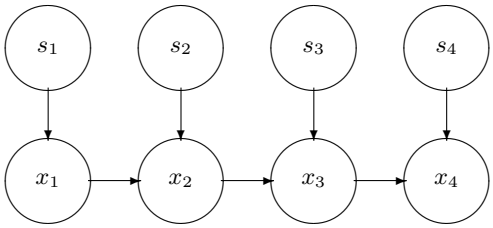


**Fig. 2**. Graphical representation of the switching LDM for speech used in this paper.

The switching LDM used in this paper assumes time dependence among the continuous $x_t$, but not among the discrete $s_t$ state variables. This is typically presented as either the graphical model of Figure 2, or as the equations

$$p(x_t, s_t|x_{t-1}) = N(x_t; A_{s_t} x_{t-1} + b_{s_t}, C_{s_t}) p(s_t)$$

$$p(x_1^T, s_1^T) = p(x_1, s_1) \prod_{t=2}^{T} p(x_t, s_t|x_{t-1})$$

If time-dependence among the discrete state variables were included, it would be analogous to modifying a GMM to become an HMM. This is an obvious improvement that is not explored in this paper.

### 2.3. Training

Training the parameters $\{A_s\}$, $\{b_s\}$ and $\{C_s\}$ is accomplished using standard EM techniques.

First, the parameters are held fixed to compute the expected of state occupancy,

$$\gamma_t^m = p(s_t = m|x_1^T).$$

Second, a new set of parameters is found that maximizes the expected log-likelihood of the data given the model. The result of this maximization step is:

$$\begin{aligned} A_m &= \left(\langle x_t x'_{t-1}\rangle_m - \langle x_t\rangle_m \langle x'_{t-1}\rangle_m\right) \cdot \\ &\quad \left(\langle x_{t-1} x'_{t-1}\rangle_m - \langle x_{t-1}\rangle_m \langle x'_{t-1}\rangle_m\right)^{-1} \\ b_m &= \langle x_t\rangle_m - A_m \langle x_{t-1}\rangle_m \\ C_M &= \left\langle (x_t - A_m x_{t-1} - b_m)(x_t - A_m x_{t-1} - b_m)' \right\rangle_m \end{aligned}$$

In these equations, we have used the shorthand notation $\langle\cdot\rangle_m$ to indicate expectation over the training data. For example,

$$\langle x_t x'_{t-1}\rangle_m = \frac{\sum_{i=1}^{T} \gamma_i^m x_t x'_{t-1}}{\sum_{i=1}^{T} \gamma_i^m}.$$

In the limit of one hidden state, the switching LDM becomes identical to the LDM, and these same equations can be used to train the $A$, $b$, and $C$ in a single pass.

## 3. OBSERVATION MODEL

The observation model relates the noisy observation to the hidden speech and noise features. The model used in this paper is the zero variance model with SNR inference, which was introduced in [3]. It is similar to several related techniques, including [4, 5, 6].

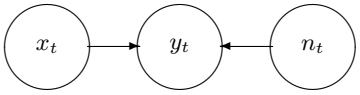


**Fig. 3**. Graphical representation of the observation model used in this paper. The observation is a non-linear function of speech and noise.

The observation model assumes that $x$ and $n$ mix linearly in the time domain, which corresponds with a non-linear mixing in the cepstral feature space.

An SNR variable $r = x - n$ is introduced to simplify calculation. If the prior distribution for speech and noise are

$$x \sim N(x; \mu_x, \Sigma_x), \quad \text{and} \quad n \sim N(n; \mu_n, \Sigma_n),$$

then the joint PDF of the noisy observation $y$ and the hidden variable $r$ can be shown to be,

$$\begin{aligned} p(y, r) &= N(y - C\ln(e^{Dr} + 1) + r; \mu_x, \Sigma_x) \\ &\quad N(y - C\ln(e^{Dr} + 1); \mu_n, \Sigma_n) \end{aligned}$$

Here, $C$ represents the matrix that rotates log mel-frequency filterbank outputs into cepstra, and $D$ represents its right inverse, such that $CD = I$.

The behavior of this joint PDF is intuitive. At high SNR, $r \gg 0$, and

$$p(y, r) \approx N(y; \mu_x, \Sigma_x) N(y - r; \mu_n, \Sigma_n).$$

That is, the observation is assumed to be clean speech, and the noise is at a level $r$ units below the observation. The converse is true for low SNR, where $r \ll 0$.

## 4. POSTERIOR ESTIMATION

The major problem with computing a posterior under the proposed system is that the switching LDM makes it computationally intractable.

The switching LDM can take any of $M$ hidden states at each time frame. As a result, for $T$ frames of speech, there are $M^T$ possible state sequences.

If the state sequence were known, the model would reduce to a time-varying LDM with known parameters, which is trivially solvable. The true posterior would contain only one mixture component.

When, as in our case, the state sequence is unknown, the exact answer is a mixture of $M^T$ components: one fore each possible state sequence.

### 4.1. Generalized Pseudo-Bayesian Technique

One approximations available for reducing the size of the search space is the generalized pseudo-Bayesian (GPB) algorithm[1]. GPB assumes that it is not important to keep track of distinct state histories whose differences occur more than $r$ frames in the past.

For $r = 1$, the posterior is collapsed into $M$ Gaussian components at each time step. Each of these components corresponds to a value the current state might take on. For $r = 2$, there are $M^2$ unique histories, and a corresponding number of reduced Gaussian components. In general, the GPB algorithm reduces the inference complexity from $M^T$ to $M^r$, where we choose $r \ll T$.

For each frame of data, three steps are performed in order: collapse, predict, and observe.

### 4.2. GPB Collapse

Before processing frame $t$, we have available $M^r$ Gaussian components corresponding to an equal number of unique state histories. Each component incorporates observations up to $y_{t-1}$ to produce a posterior for $x_{t-1}$.

$$q(x_{t-1}, y_1^{t-1}, s_{t-r}^{t-1})$$

The posterior is marginalized over states occurring $r$ frames in the past. Each sum of Gaussians is approximated by a single Gaussian component using moment matching. This collapses together all Gaussians that share a history of length $r-1$.

$$\begin{aligned} & q(x_{t-1}, y_1^{t-1} | s_{t-r+1}^{t-1}) \\ \approx & \sum_{s_{t-r}} p(x_{t-1}, y_1^{t-1} | s_{t-r+1}^{t-1}) p(s_{t-r}) \end{aligned}$$

### 4.3. GPB Predict

The second, prediction, step is to branch out each remaining hypotheses $M$ times, once for each possible state $s_t$.

$$\begin{aligned} & q(x_t, y_1^{t-1} | s_{t-r+1}^{t}) \\ = & \int q(x_{t-1}, y_1^{t-1} | s_{t-r+1}^{t-1}) p(x_t | x_{t-1}, s_t) d\, x_{t-1} \end{aligned}$$

Because all of the distributions are Gaussian, the marginalization is trivial.

### 4.4. GPB Observe

At this point, we have $M^r$ components that describe $x_t$, but the current observation $y_t$ has not been accounted for. Incorporating the current observation allows us to produce a posterior distribution for $x_t$ that includes all observations up to and including $y_t$.

Because the observation model is non-linear, we must use the approximation from Section 3. The prior distribution for the hidden variables comes from the output of the prediction step. The output of the approximate observation model is the posterior distribution,

$$q(x_t, y_1^t | s_{t-r+1}^t)$$

### 4.5. Enhancement

In addition to serving as the input for the next frame to process, this approximate Gaussian posterior can also be used to produce estimates of the moments of $x_t$. It is these moments that we use to perform noise robust recognition. The MMSE estimate $E[x_t | y_1^t]$ can be fed directly to a traditional recognition system, or augmented with the second moment for use with uncertainty decoding[7].

$$\begin{aligned} q(x_t, y_1^t) &= \sum_{s_{t-r+1}^t} q(x_t, y_1^t | s_{t-r+1}^t) p(s_{t-r+1}^t) \\ E[x_t | y_1^t] &\approx \frac{\int x_t q(x_t, y_1^t) d\, x_t}{\int q(x_t, y_1^t) d\, x_t} \\ E[(x_t)^2 | y_1^t] &\approx \frac{\int (x_t)^2 q(x_t, y_1^t) d\, x_t}{\int q(x_t, y_1^t) d\, x_t} \end{aligned}$$

## 5. RESULTS

The experiments presented here were conducted using the data, code, and training scripts provided within the original Aurora 2 task[8].

The Aurora 2 task consists of recognizing strings of English digits embedded in a range of artificial noise conditions. Although the framework provides for evaluation against many noise conditions under different training strategies, we present here results for test set A with clean acoustic model training only. The objective of the current experiments is to test the feasibility and behavior of the approximation method before moving to more complex inference algorithms.

The acoustic model used for recognition is the standard "complex back-end" trained on uncorrupted, unprocessed data. It contains eleven 16-state whole-word models, in addition to the "sil" and "sp" models. Each state consists of 20 diagonal Gaussian mixture components.

To conform with our observation models, the feature generation was modified slightly from the reference implementation. In particular, we replaced the log energy feature with $c_0$, and changed from using spectral magnitude to using power spectral density as the input to the mel-frequency filterbank.

All of the experiments use the same set of global speech models. Eight models were built using the procedure outlined in Section 2. Each model contained between 1 and 128 hidden states.

For each utterance, an utterance-specific noise model is built. It consists of a single Gaussian mixture component, with parameters trained on the first and last ten frames of the noisy utterance. The model assumes that the noise frames are independent over time. Since our algorithms also produce posterior noise estimates,

| M | Subway | Babble | Car | Exhibition | Ave. |
|---|---|---|---|---|---|
| 0 | 65.75 | 43.17 | 57.54 | 67.84 | 58.58 |
| 1 | 73.10 | 61.58 | 80.09 | 71.57 | 71.59 |
| 2 | 74.74 | 64.19 | 81.81 | 73.55 | 73.57 |
| 4 | 80.40 | 65.11 | 85.42 | 76.90 | 76.96 |
| 8 | 80.47 | 66.88 | 86.06 | 78.11 | 77.88 |
| 16 | 80.55 | 67.52 | 86.19 | 77.82 | 78.02 |
| 32 | 83.20 | 69.63 | 86.98 | 79.25 | 79.76 |
| 64 | 83.57 | 68.84 | 87.41 | 79.24 | 79.76 |
| 128 | 83.70 | 69.69 | 87.36 | 79.12 | 79.97 |

**Table 1**. Accuracy on Aurora 2, Set A. Results are average across 0 dB to 20 dB conditions. Enhancement is performed in forward direction only.

| M | Subway | Babble | Car | Exhibition | Ave. |
|---|---|---|---|---|---|
| 0 | 65.75 | 43.17 | 57.54 | 67.84 | 58.58 |
| 1 | 76.52 | 68.26 | 83.87 | 76.18 | 76.21 |
| 2 | 77.04 | 70.05 | 84.46 | 76.53 | 77.02 |
| 4 | 80.91 | 69.35 | 86.51 | 77.61 | 78.60 |
| 8 | 81.40 | 70.95 | 87.22 | 79.39 | 79.74 |
| 16 | 81.55 | 71.31 | 87.52 | 79.42 | 79.95 |
| 32 | 83.59 | 72.36 | 87.82 | 80.19 | 80.99 |
| 64 | 84.06 | 72.07 | 88.28 | 80.33 | 81.18 |
| 128 | 84.14 | | | | |

**Table 2**. Accuracy on Aurora 2, Set A. Results are average across 0 dB to 20 dB conditions. Forward and backward enhancement are combined.

it is conceivable that the noise model could be adapted with an on-line EM algorithm.

The enhancement algorithm was run with the history parameter $r = 1$, which makes it run almost at the same speed as an equivalent system built with a GMM or HMM.

### 5.1. Causal Enhancement

The enhancement algorithm, as described in the previous section, was run on each utterance of the test set. Enhancement of frame $t$ used knowledge of data up to and including frame $t$. There was no lookahead. The results are listed in Table 1.

The "Car" noise type benefits the most from this algorithm. It is also the noise type that is most stationary, and therefore matches the modeling assumptions well.

### 5.2. Non-causal Enhancement

The causal enhancement algorithm tended to suppress the beginning of each word, but not the end. This was usually accompanied by a high second order central moment in the posterior. This is consistent with the algorithm being unsure whether it was seeing the beginning of the utterance, or slightly more energetic noise.

To overcome this problem, we ran the entire utterance through the algorithm twice. The first time, just as before, produced a posterior from running the signal through in the forward direction: $q(x|\mu_F, \Sigma_F)$. The second pass was run by passing the input through in reverse time order: $q(x|\mu_B, \Sigma_B)$. The two passes were combined using the heuristic,

$$\mu = (\Sigma_F^{-1} + \Sigma_B^{-1})^{-1}(\Sigma_F^{-1}\mu_F + \Sigma_B^{-1}\mu_B).$$

This has the desired result of choosing $\mu_B$ when $\Sigma_F$ is large, and choosing $\mu_F$ when $\Sigma_B$ is large. Results are shown in Table 2. The improvement is noticeable, especially for very small model sizes.

## 6. SUMMARY

This paper has presented a unified, nonlinear, non-stationary, stochastic model for estimating and removing the effects of background noise on speech cepstra. The model is the union of dynamic system equations for speech and noise, and a model describing how speech and noise are mixed.

Preliminary results have indicated that this model can reduce digit error rate, even with relatively small number of mixture components.

To expand upon this initial result, future work should include:

- Increasing $r$ to more closely approximate the true posterior distribution.
- Modeling the linear dynamics of noise in addition to speech.
- Augmenting the switching LDM with discrete state transition probabilities.
- Exploring other approximation strategies for this system.

## 7. REFERENCES


[1] Y. Bar-Shalom and X.-R. Li, *Estimation and Tracking: Principles, Techniques, and Software*, Artech House, Norwood, MA., 1993.

[2] R. Singh and B. Raj, "Tracking noise via dynamical systems with a continuum of states," in *Proc. ICASSP*, 2003, vol. I, pp. 396–399.

[3] J. Droppo, L. Deng, and A. Acero, "A comparison of three non-linear observation models for noisy speech features," in *Proc. 2003 Eurospeech*, Geneva, Switzerland, September 2003, pp. 681–684.

[4] P. Moreno, *Speech Recognition in Noisy Environments*, Ph.D. thesis, Carnegie Mellon University, 1996.

[5] B. Frey, L. Deng, A. Acero, and T. Kristjansson, "ALGONQUIN: Iterating Laplace's method to remove multiple types of acoustic distortion for robust speech recognition," in *Proc. 2001 Eurospeech*, Aalbork, Denmark, September 2001.

[6] V. Stouten, H. Van hamme, K. Demuynck, and P. Wambacq, "Robust speech recognition using model-based feature enhancement," in *Proc. 2003 Eurospeech*, Geneva, Switzerland, September 2003, pp. 17–20.

[7] J. Droppo, A. Acero, and L. Deng, "Uncertainty decoding with SPLICE for noise robust speech recognition," in *Proc. 2002 ICASSP*, Orlando, Florida, May 2002.

[8] H. G. Hirsch and D. Pearce, "The AURORA experimental framework for the performance evaluations of speech recognition systems under noisy condidions," in *ISCA ITRW ASR2000 "Automatic Speech Recognition: Challenges for the Next Millennium"*, Paris, France, September 2000.