CAMBRIDGE UNIVERSITY

ENGINEERING DEPARTMENT

Joint Uncertainty Decoding for Robust Large Vocabulary Speech Recognition

H. Liao and M.J.F. Gales

CUED/F-INFENG/TR.552
November 8, 2006

Cambridge University Engineering Department
Trumpington Street
Cambridge. CB2 1PZ
England

E-mail: hl251@eng.cam.ac.uk
http://mi.eng.cam.ac.uk/~hl251

Abstract

Standard techniques to increase automatic speech recognition noise robustness typically assume recognition models are clean trained. This "clean" training data may in fact not be clean at all, but may contain channel variations, varying noise conditions, as well as different speakers. Hence rather than considering noise robustness techniques as compensating clean acoustic models for environmental noise, they may be thought of as reducing the acoustic mismatch between training and test conditions. This report examines the application of VTS model compensation or model-based `Joint` uncertainty decoding to clean and multistyle trained systems. An EM-based noise estimation procedure is also presented to produce ML VTS or `Joint` noise models depending on the form of compensation used. Alternatively, compared to multistyle training, adaptive training with `Joint` uncertainty transforms, also referred to as JAT in this work, provides a better method for handling heterogeneous data. With JAT, the uncertainty bias added to the model variances de-weights observations proportional to the noise level. In this way, `Joint` transforms normalise the noise from the data allowing the canonical model to solely represent the underlying "clean" acoustic signal. This report presents a novel `Joint` adaptive training framework including formula for estimating the transforms and canonical model parameters. Lastly, large vocabulary systems are often trained on multistyle data sets such as broadcast news or conversational telephone speech that have a variety of noise conditions. However, to date not much research has been done on compensating such systems built with non-artificially corrupted data. In this report, experiments are conducted on an artificially corrupted Resource Management database and the large vocabulary Broadcast News corpus of collected broadcast recordings.

1 Introduction

Many standard noise robustness techniques presume that the acoustic model has been trained on clean audio data. Though the signal-to-noise ratio may be high, there still may be residual background noise, channel differences and likely many different speakers. Hence, clean training may be considered a form of multistyle training [28] since acoustic models not only model speech variability, but also additional variation due to different voices and interfering noises. Multistyle training generates acoustic models that have shown to be more robust to noise than those produced from clean data [18]. Furthermore, it has been demonstrated that front-end feature enhancement schemes are more effective with multistyle acoustic models rather than clean trained [8, 40]; these results are reported on artificially corrupted data with small and 5k-word vocabularies. To date, there has been little research on applying noise robustness techniques to large vocabulary systems trained on non-artificially corrupted data. This report explores the use of powerful noise robustness methods such as VTS model compensation [30] and model-based `Joint` uncertainty decoding [25] on such a system. An EM-based noise estimation procedure is presented to produce ML VTS or `Joint` models of the convolutional and additive noise depending on the form of compensation used. This provides a better estimate of the noise, given the compensation form, and allows for estimation of noise even during speech, as opposed to the frequently used technique of estimating the noise from background, non-speech areas as in [9, 3].

An alternative form of model training to multistyle is adaptive training. This may be applied to remove unwanted factors, such as speaker differences or the acoustic environment, from being included in the acoustic models [2, 5, 16]. Rather than force the acoustic model to represent all these factors, as expected in multistyle training, transforms are used to model the variation from different factors. A standard form of adaptive training is to use MLLR transforms [2]; however, due to their linear nature they can only normalise low levels of noise, and hence are unsuitable for adaptive training with data that has large variations in SNR. This motivates a novel model training framework called `Joint` adaptive training (JAT), based on noise normalisation using `Joint` transforms for training models on noisy data. In contrast to adaptive training with CMLLR [14], JAT takes into account the SNR of the data when estimating the canonical model parameters. When the noise subsumes the speech, the uncertainty variance bias ensures those observations do not contribute to the model parameter update. In this way, JAT weights the training data using uncertainty due to noise. Hence, JAT explicitly handles a large range of SNR in the training data, producing a final acoustic model that is truly clean.

This report is organised as follows: First a model of the noisy environment is introduced along with a general framework for representing speech in noise. Two compensation forms that fall within this framework are described in section 3: `Joint` uncertainty decoding and VTS compensation. Section 4 presents an EM framework to estimate a ML noise model for generating `Joint` and VTS compensation parameter. Following in section 5, the `Joint` adaptive training framework is introduced. Results are reported in section 6 on an artificially corrupted Resource Management database [34] and a simplified Broadcast New system [19] that transcribes actual collected broadcast recordings. Lastly, conclusions and future directions are given in section 7.

2 Modeling Environmental Noise

A variety of factors, which affect the end speech signal that is transcribed, may be considered noise. There may be ambient noise around the speaker, the speaker themself might make breathy or lip sounds, task or emotional stress may contribute changes in the speech, or the channel might be noisy or different. It is the first and last factors that are typically the dominant sources of

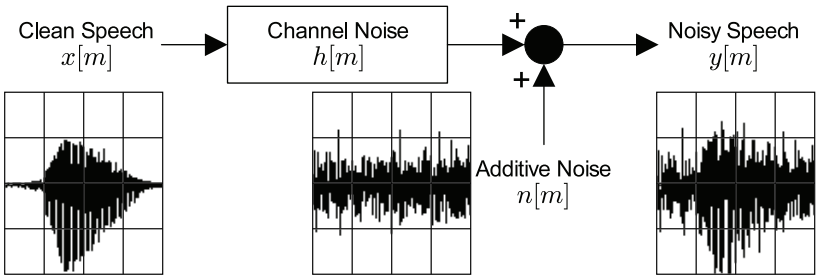

Figure 1: Model of the corrupted speech environment.

noise, yielding this effective and often used model of the noisy acoustic environment shown in figure 1 [12, 1]. This depicts the following model of corrupted speech

$$y[m] = h[m] * x[m] + n[m] \tag{1}$$

where $y[m]$ is the noise-corrupted speech, $h[m]$ represents the channel or convolutional noise, $x[m]$ the clean speech and $n[m]$ the additive noise. In the cepstral domain this relationship is given by

$$y_i = \boldsymbol{c}_i \log(\exp(\boldsymbol{C}^{-1}(\boldsymbol{x} + \boldsymbol{h})) + \exp(\boldsymbol{C}^{-1}\boldsymbol{n})) \tag{2}$$

$$= x_i + h_i + \boldsymbol{c}_i \log(\boldsymbol{1} + \exp(\boldsymbol{C}^{-1}(\boldsymbol{n} - \boldsymbol{x} - \boldsymbol{h}))) \tag{3}$$

where matrices $\boldsymbol{C}$ and $\boldsymbol{C}^{-1}$ are the discrete cosine transform matrix (DCT) and its inverse. The vector $\boldsymbol{c}_i$ denotes the i^{th} row of the DCT. The log and exp functions operate at an element level on the resultant filterbank vectors.

A Monte Carlo simulation can give insight as to how noise affects a clean speech distribution. This is shown in figure 2, where the effects of increasing the magnitude of noise with variance 1 are demonstrated on a clean speech distribution of mean 10, variance 5, in the log-spectral domain. When the SNR is high, there is a distinct bimodal distribution. However, as the noise energy increases, the separability is lost and the distribution is once again unimodal with a strong skew. Also, there is a increase in the mean of the noisy speech and a decrease in variance which can be discerned by observing how the estimated ML normal distribution changes (dashed). Eventually when the noise mean is sufficiently large, the clean speech is subsumed, and the corrupted speech distribution becomes the noise distribution.

One framework for incorporating the effects of environmental noise is represented in the dynamic Bayesian network (DBN) shown in figure 3. Here, the noise corrupted speech observation $\boldsymbol{o}_t$ at time t is assumed to be conditionally independent of all other observations given the hidden clean speech $\boldsymbol{s}_t$ and noise $\boldsymbol{n}_t$ at that time frame. The clean speech and noise are assumed to be generated by HMMs with states θ_t^n for the noise[1] and θ_t for the clean speech. In most automatic speech recognition systems, dynamic features are appended to the static: For example in a sequence of observations $\boldsymbol{O} = \{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_T\}$, a single observation is comprised of static, velocity and acceleration features $\boldsymbol{o}_t = [\boldsymbol{y}_t^\mathsf{T} \;\; \Delta\boldsymbol{y}_t^\mathsf{T} \;\; \Delta^2\boldsymbol{y}_t^\mathsf{T}]^\mathsf{T}$ and the hidden clean speech vector defined as $\boldsymbol{s}_t = [\boldsymbol{x}_t^\mathsf{T} \;\; \Delta\boldsymbol{x}_t^\mathsf{T} \;\; \Delta^2\boldsymbol{x}_t^\mathsf{T}]^\mathsf{T}$.

[1] A single state is assumed for the noise model in this report.

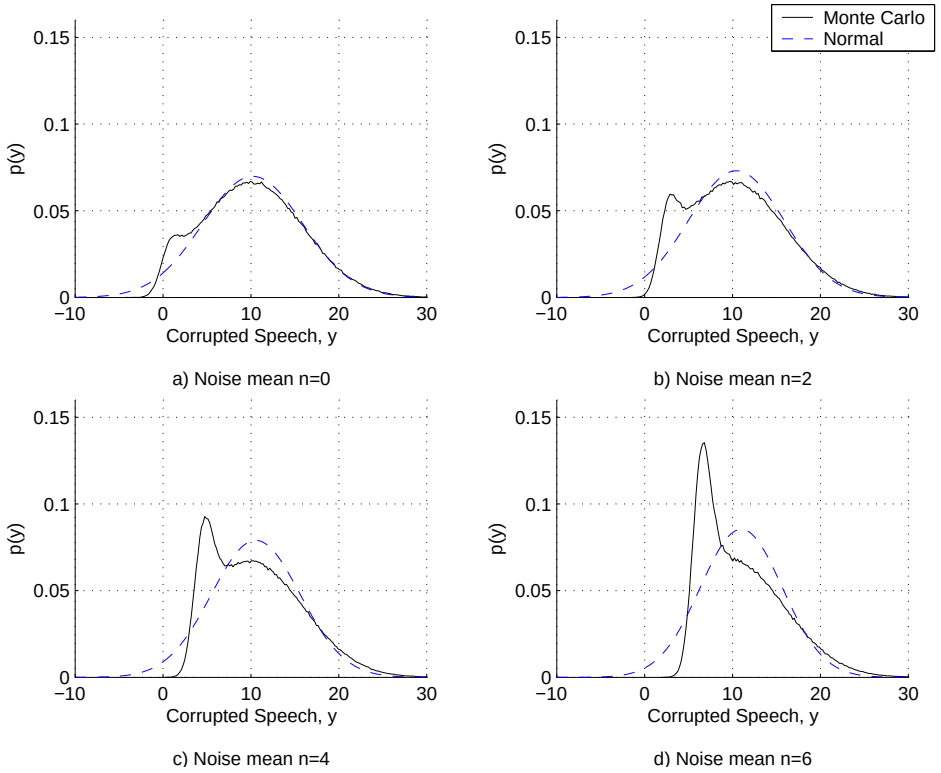

Figure 2: Monte Carlo simulation of effect of increasing noise on speech.

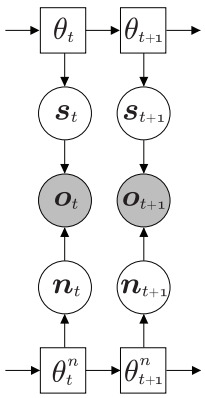

Figure 3: Noise robustness DBN. Emitting states shaded.

Using this model of the corrupted speech depicted in figure 3, the likelihood of the corrupted speech may be expressed as

$$p(\boldsymbol{o}_t|\mathcal{M}, \check{\mathcal{M}}, \boldsymbol{\theta}_t) = \int p(\boldsymbol{o}_t|\boldsymbol{s}_t, \check{\mathcal{M}}) p(\boldsymbol{s}_t|\mathcal{M}, \theta_t) d\boldsymbol{s}_t \tag{4}$$

where

$$p(\boldsymbol{o}_t|\boldsymbol{s}_t, \check{\mathcal{M}}) = \int p(\boldsymbol{o}_t|\boldsymbol{s}_t, \boldsymbol{n}_t) p(\boldsymbol{n}_t|\check{\mathcal{M}}, \theta_t^n) d\boldsymbol{n}_t \tag{5}$$

and $\check{\mathcal{M}}$ denoting the compensation model parameters, which may or may not have an explicit model of the noise[2]. The uncompensated "clean" acoustic model $\mathcal{M}$ consists of Gaussian components each defined by a prior, c_m, mean, $\boldsymbol{\mu}_s^{(m)}$, and variance, $\boldsymbol{\Sigma}_s^{(m)}$. Thus, the likelihood calculation in equation 4 has two distinct parts: Only the first $p(\boldsymbol{o}_t|\boldsymbol{s}_t, \check{\mathcal{M}})$, is a function of the noise; the second is the standard clean speech likelihood.

This noise robustness DBN has been used as a framework for uncertainty decoding [7, 24]. In the uncertainty decoding version of `SPLICE`, Bayes' rule is applied to equation 5 and a piece-wise linear, `SPLICE`, approximation of the clean speech posterior is used [7]. The front-end `Joint` decoding scheme directly estimates equation 5 using the joint distribution of the clean and corrupted speech. In both forms, the compensation parameters $\check{\mathcal{M}}$ are independent of the model parameters $\mathcal{M}$ allowing fast front-end compensation. More powerful schemes may maintain dependence of the corrupted speech conditional distribution to the acoustic model, permitting the compensation to operate more informatively using a detailed knowledge of the clean speech model.

[2]If the compensation model parameters $\check{\mathcal{M}}$ are single-pass retrained, as in [24], then no noise model is explicitly estimated.

3 Model-based Noise Compensation

Many approaches have been formulated to improve automatic noisy speech recognition. These can be broadly grouped into the following areas: inherently robust front-ends, feature enhancement, and model-based compensation. Inherently robust front-ends such as RASTA [17] or other perceptually motivated auditory processing attempt to provide front-end forms that are immune to noise in speech; while they may provide some small level of robustness, they are generally not very effective [37, 15]. Hence most noise robustness schemes aim to directly address the effects of noise on speech as given by equation 3; this assumes that the frame/state alignments do not differ between clean and corrupted speech. Enhancement schemes, such as spectral subtraction [4] or cepstral mean normalisation, aim to remove the corrupting noise itself from the features, but cannot effectively address the varying compressive effect of the noise on model variances [24]. More powerful model-based techniques approximate equation 3 to compensate the mean and variance of the state output distributions based on a model of the noise. Two specific model-based techniques are discussed in this report: first-order VTS model compensation and model-based `Joint` Uncertainty Decoding.

3.1 Vector Taylor Series Compensation

Deriving a corrupted speech output distribution, given the clean acoustic model and a noise model, is not straightforward. Directly determining the expected value of equation 3 is problematic due to the non-linear effect of noise on cepstral speech features. Hence many approximations to this function have been proposed, such as selecting the maximum of either the noise or speech, i.e. noise masking [43] or Parallel Model Combination [12]. Another approach is to linearise equation 3 with a truncated vector Taylor series (VTS) [30, 21, 1] to individually update each model component. The first-order VTS approximation of the static corrupted speech for dimension i is

$$y_{vts,i} = y_i\big|_{\mu_0} + \boldsymbol{\nabla}_{\mathbf{x}}\, y_i\big|_{\mu_0} \cdot \left(\boldsymbol{x} - \boldsymbol{\mu}_x^{(m)}\right) + \boldsymbol{\nabla}_{\mathbf{n}}\, y_i\big|_{\mu_0} \cdot (\boldsymbol{n} - \boldsymbol{\mu}_n) + \boldsymbol{\nabla}_{\mathbf{h}}\, y_i\big|_{\mu_0} \cdot (\boldsymbol{h} - \boldsymbol{\mu}_h) \tag{6}$$

where $\big|_{\mu_0}$ is the Taylor series expansion point indicating the function is evaluated at the clean speech component mean $\boldsymbol{\mu}_x^{(m)}$, and current estimates of the additive noise mean $\boldsymbol{\mu}_n$ and channel noise $\boldsymbol{\mu}_h$. The symbol $\cdot$ indicates the dot product. Hence the corrupted speech mean may be approximated by the expected value of equation 6

$$\begin{aligned}
\mu_{y,i}^{(m)} &= \mathcal{E}\left\{y_i\right\} && (7)\\
&\approx \mathcal{E}\left\{y_{vts,i}\right\} = y_i\big|_{\mu_0} && (8)\\
&= \mu_{x,i}^{(m)} + \mu_{h,i} + \boldsymbol{c}_i \log(\mathbf{1} + \exp(\boldsymbol{C}^{\text{-}1}(\boldsymbol{\mu}_n - \boldsymbol{\mu}_x^{(m)} - \boldsymbol{\mu}_h))) && (9)
\end{aligned}$$

assuming the clean speech, additive noise and channel noise are independent of each other and are Gaussian distributed random variables. The covariance is given by

$$\begin{aligned}
\boldsymbol{\Sigma}_y^{(m)} &= \mathcal{E}\left\{\boldsymbol{y}\boldsymbol{y}^{\mathsf{T}}\right\} - \boldsymbol{\mu}_y^{(m)}\boldsymbol{\mu}_y^{(m)\mathsf{T}} && (10)\\
&\approx \mathcal{E}\left\{\boldsymbol{y}_{vts}\boldsymbol{y}_{vts}^{\mathsf{T}}\right\} - \boldsymbol{\mu}_y^{(m)}\boldsymbol{\mu}_y^{(m)\mathsf{T}} && (11)\\
&\approx \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0} \boldsymbol{\Sigma}_x^{(m)} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0}^{\mathsf{T}} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0} \boldsymbol{\Sigma}_h \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0}^{\mathsf{T}} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} \boldsymbol{\Sigma}_n \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}^{\mathsf{T}} && (12)
\end{aligned}$$

which also assumes the clean speech and noise are random variables and independent. Since the Jacobian matrices $\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}$, $\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}$ and $\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}$ are full, the corrupted speech covariance matrix will also be full and hence is normally diagonalised for standard decoders. Also, it is often assumed that the channel noise does not vary, that is $\boldsymbol{\Sigma}_h = 0$. Hence the static corrupted speech variance may be given by

$$\boldsymbol{\Sigma}_y^{(m)} \approx \mathtt{diag}\left\{ \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0} \boldsymbol{\Sigma}_x^{(m)} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0}^{\mathsf{T}} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} \boldsymbol{\Sigma}_n \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}^{\mathsf{T}} \right\} \tag{13}$$

Like with PMC, using these update formula assumes that a clean speech Gaussian component corrupted by noise may be approximated by another Gaussian distribution; this is clearly not optimal since it was shown in figure 2 that the corrupted speech distribution can be bimodal. Nevertheless, for efficiency this approximation is maintained.

The $D_s \times D_s$ sized Jacobian matrices, where D_s is the number of static features, are given by

$$\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0} = \left[\boldsymbol{\nabla}_{\mathbf{x}}\, y_i\Big|_{\mu_0} \cdots \boldsymbol{\nabla}_{\mathbf{x}}\, y_{D_s}\Big|_{\mu_0}\right]^{\mathsf{T}} = \boldsymbol{I} - \boldsymbol{C}\boldsymbol{F}\boldsymbol{C}^{-1} \tag{14}$$

$$\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0} = \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0}, \qquad \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} = \boldsymbol{C}\boldsymbol{F}\boldsymbol{C}^{-1} \tag{15}$$

and the elements of the diagonal matrix $\boldsymbol{F}$ are

$$f_{ii} = \frac{\exp(\boldsymbol{c}_i^{-1}(\boldsymbol{\mu}_n - \boldsymbol{\mu}_x^{(m)} - \boldsymbol{\mu}_h))}{1 + \exp(\boldsymbol{c}_i^{-1}(\boldsymbol{\mu}_n - \boldsymbol{\mu}_x^{(m)} - \boldsymbol{\mu}_h))} \tag{16}$$

The terms f_{ii} vary from 0 to 1 depending on the ratio of the speech to the noise. If the noise level $\boldsymbol{\mu}_n$ is greater than the speech $\boldsymbol{\mu}_x^{(m)}$ in the log-spectral domain, then $f_{ii} \to 1$ and $\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\big|_{\mu_0}$ tends to zero; otherwise if little noise is present, $f_{ii} \to 0$ and $\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\big|_{\mu_0}$ also tends to identity. The term $\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\big|_{\mu_0}$ behaves in the opposite manner to $\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\big|_{\mu_0}$. From equations 14 and 15 it can be shown that

$$\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} = \boldsymbol{I} \tag{17}$$

Hence it can be observed that the compensated variance in equation 13 scales between the clean speech variance in low noise to the additive noise variance in high noise.

3.1.1 Compensating Dynamic Features

Standard acoustic models use simple differences or linear regression to compute delta parameters to model the dynamic features of speech. This complicates the compensation of these features for noisy conditions. Hence a Continuous-Time approximation [12] may be used to derive the compensated dynamic parameters. Full derivations for the dynamic features are given in appendix A, with the final compensation formulae summarised here. The delta noisy mean may be approximated by

$$\boldsymbol{\mu}_{\Delta y}^{(m)} = \mathcal{E}\left\{\frac{\partial \boldsymbol{y}}{\partial t}\right\} = \mathcal{E}\left\{\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\frac{\partial \boldsymbol{x}}{\partial t} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\frac{\partial \boldsymbol{n}}{\partial t} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\frac{\partial \boldsymbol{h}}{\partial t}\right\} \tag{18}$$

$$\approx \mathcal{E}\left\{\frac{\partial \boldsymbol{y}_{vts}}{\partial \boldsymbol{x}}\frac{\partial \boldsymbol{x}}{\partial t} + \frac{\partial \boldsymbol{y}_{vts}}{\partial \boldsymbol{n}}\frac{\partial \boldsymbol{n}}{\partial t} + \frac{\partial \boldsymbol{y}_{vts}}{\partial \boldsymbol{h}}\frac{\partial \boldsymbol{h}}{\partial t}\right\} \tag{19}$$

$$\approx \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0} \boldsymbol{\mu}_{\Delta x}^{(m)} \tag{20}$$

assuming that the additive and convolutional noise are constant, i.e. $\Delta \boldsymbol{n} = \Delta \boldsymbol{h} = 0$. Similarly for the dynamic noisy speech variance

$$\boldsymbol{\Sigma}_{\Delta y}^{(m)} = \mathcal{E}\left\{\frac{\partial \boldsymbol{y}}{\partial t}\frac{\partial \boldsymbol{y}}{\partial t}^{\mathsf{T}}\right\} - \boldsymbol{\mu}_{\Delta y}^{(m)}\boldsymbol{\mu}_{\Delta y}^{(m)\mathsf{T}} \tag{21}$$

$$\approx \mathcal{E}\left\{\frac{\partial \boldsymbol{y}_{vts}}{\partial t}\frac{\partial \boldsymbol{y}_{vts}}{\partial t}^{\mathsf{T}}\right\} - \boldsymbol{\mu}_{\Delta y}^{(m)}\boldsymbol{\mu}_{\Delta y}^{(m)\mathsf{T}} \tag{22}$$

$$\approx \mathtt{diag}\left\{\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0} \boldsymbol{\Sigma}_{\Delta x}^{(m)} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0}^{\mathsf{T}} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} \boldsymbol{\Sigma}_{\Delta n} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}^{\mathsf{T}}\right\} \tag{23}$$

assuming that there is no variance in the convolutional noise. Following the same reasoning, the delta-delta parameters are then

$$\boldsymbol{\mu}_{\Delta^2 y}^{(m)} \approx \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0} \boldsymbol{\mu}_{\Delta^2 x}^{(m)} \tag{24}$$

$$\boldsymbol{\Sigma}_{\Delta^2 y}^{(m)} \approx \mathtt{diag}\left\{ \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0} \boldsymbol{\Sigma}_{\Delta^2 x}^{(m)} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0}^{\mathsf{T}} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} \boldsymbol{\Sigma}_{\Delta^2 n} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}^{\mathsf{T}} \right\} \tag{25}$$

although more approximations are required as detailed in section A.1.

3.2 `Joint` Uncertainty Decoding

Uncertainty decoding for noise compensation has shown to be an effective compromise between fast, feature enhancement schemes and more computationally intensive model-based schemes [7, 24, 25]. Uncertainty decoding takes advantage of the factorisation given in equation 4 by using an appropriate form of approximation for the conditional distribution in equation 5 for a particular noise environment. As the complexity of this approximation may be independent of the complexity of the actual acoustic models, there is a large degree of flexibility in choosing the computational cost of the decoding process. In front-end uncertainty decoding, conditional distributions $p(\boldsymbol{o}_t|\boldsymbol{s}_t, \check{\mathcal{M}})$ are estimated for different regions of the corrupted acoustic space; it was found that this approach suffered from a significant problem [26]. In contrast, the more efficient model-based `Joint` uncertainty decoding form [25, 26] ties the corrupted speech conditional distribution given the clean to the acoustic model distribution. The components m in the acoustic model can be clustered into classes of acoustically similar components [36, 13]. Hence, the conditional is a function of which class, r_m, the acoustic model component belongs to

$$p(\boldsymbol{o}_t|\boldsymbol{s}_t, \check{\mathcal{M}}) \approx p(\boldsymbol{o}_t|\boldsymbol{s}_t, r_m, \check{\mathcal{M}}) \tag{26}$$

This clean speech given the corrupted speech conditional can be derived from a joint distribution of the clean and corrupted speech for the class r_m. The joint distribution is assumed to be Gaussian

$$\begin{bmatrix} \boldsymbol{s}_t \\ \boldsymbol{o}_t \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu}_s^{(r_m)} \\ \boldsymbol{\mu}_o^{(r_m)} \end{bmatrix}, \begin{bmatrix} \boldsymbol{\Sigma}_s^{(r_m)} & \boldsymbol{\Sigma}_{so}^{(r_m)} \\ \boldsymbol{\Sigma}_{os}^{(r_m)} & \boldsymbol{\Sigma}_o^{(r_m)} \end{bmatrix} \right) \tag{27}$$

and thus the conditional distribution will also be Gaussian. Although, it has been shown that this conditional is highly non-Gaussian [3, 24], results indicate this is not an unsuitable approximation [26]. When this form is used in the uncertainty decoding framework, the corrupted speech observation likelihood may be given by

$$p(\boldsymbol{o}_t|\theta_t, \mathcal{M}, \check{\mathcal{M}}) = \sum_{m \in \theta_t} c_m |\boldsymbol{A}^{(r_m)}| \mathcal{N}\left(\boldsymbol{A}^{(r_m)} \boldsymbol{o}_t + \boldsymbol{b}^{(r_m)}; \boldsymbol{\mu}_s^{(m)}, \boldsymbol{\Sigma}_s^{(m)} + \boldsymbol{\Sigma}_{\mathsf{b}}^{(r_m)} \right) \tag{28}$$

for state θ_t and the transform parameters

$$\begin{aligned} \boldsymbol{A}^{(r_m)} &= \boldsymbol{\Sigma}_s^{(r_m)} \boldsymbol{\Sigma}_{os}^{(r_m)\text{-}1}, \qquad \boldsymbol{b}^{(r_m)} = \boldsymbol{\mu}_s^{(r_m)} - \boldsymbol{A}^{(r_m)} \boldsymbol{\mu}_o^{(r_m)} \\ \boldsymbol{\Sigma}_{\mathsf{b}}^{(r_m)} &= \boldsymbol{A}^{(r_m)} \boldsymbol{\Sigma}_o^{(r_m)} \boldsymbol{A}^{(r_m)\mathsf{T}} - \boldsymbol{\Sigma}_s^{(r_m)} \end{aligned} \tag{29}$$

So that full covariance decoding is not necessary, a diagonal approximation of the joint distribution, given by equation 27, may be made

$$\mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu}_s^{(r_m)} \\ \boldsymbol{\mu}_o^{(r_m)} \end{bmatrix}, \begin{bmatrix} \boldsymbol{\Sigma}_s^{(r_m)} & \boldsymbol{\Sigma}_{so}^{(r_m)} \\ \boldsymbol{\Sigma}_{os}^{(r_m)} & \boldsymbol{\Sigma}_o^{(r_m)} \end{bmatrix}\right) \approx$$
$$\mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu}_x^{(r_m)} \\ \boldsymbol{\mu}_{\Delta x}^{(r_m)} \\ \boldsymbol{\mu}_{\Delta^2 x}^{(r_m)} \\ \boldsymbol{\mu}_y^{(r_m)} \\ \boldsymbol{\mu}_{\Delta y}^{(r_m)} \\ \boldsymbol{\mu}_{\Delta^2 y}^{(r_m)} \end{bmatrix}, \begin{bmatrix} \begin{bmatrix} \boldsymbol{\Sigma}_x^{(r_m)} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma}_{\Delta x}^{(r_m)} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \boldsymbol{\Sigma}_{\Delta^2 x}^{(r_m)} \end{bmatrix} & \begin{bmatrix} \boldsymbol{\Sigma}_{xy}^{(r_m)} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma}_{\Delta xy}^{(r_m)} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \boldsymbol{\Sigma}_{\Delta^2 xy}^{(r_m)} \end{bmatrix} \\ \begin{bmatrix} \boldsymbol{\Sigma}_{xy}^{(r_m)} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma}_{\Delta xy}^{(r_m)} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \boldsymbol{\Sigma}_{\Delta^2 xy}^{(r_m)} \end{bmatrix}^{\mathsf{T}} & \begin{bmatrix} \boldsymbol{\Sigma}_y^{(r_m)} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma}_{\Delta y}^{(r_m)} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \boldsymbol{\Sigma}_{\Delta^2 y}^{(r_m)} \end{bmatrix} \end{bmatrix}\right) \tag{30}$$

where the covariances are assumed diagonal, and the covariances between the static, velocity and accelerations are considered zero. This diagonal approximation was shown to provide good robustness, although block-diagonal and full covariance forms, which require full covariance decoding, showed even better results; keeping the linear feature transform full, and diagonalising the variance bias term produced poor results [25].

Increasing the number of classes R to equal the number of model components M, using this diagonal acoustic model variance approximation $\boldsymbol{\Sigma}_{x,\texttt{diag}}^{(m)}$, is equivalent to VTS model compensation of each individual acoustic model component. The component corrupted speech likelihood in equation 28 may be re-expressed

$$p(\boldsymbol{o}_t|\theta_t, \mathcal{M}, \check{\mathcal{M}}, m) = |\boldsymbol{A}^{(r_m)}|\mathcal{N}\left(\boldsymbol{A}^{(r_m)}\boldsymbol{o}_t + \boldsymbol{b}^{(r_m)}; \boldsymbol{\mu}_s^{(m)}, \boldsymbol{\Sigma}_s^{(m)} + \boldsymbol{\Sigma}_{\mathsf{b}}^{(r_m)}\right) \tag{31}$$
$$= \mathcal{N}\big(\boldsymbol{o}_t; \boldsymbol{A}^{(r_m)\text{-}1}\big(\boldsymbol{\mu}_s^{(m)} - \boldsymbol{\mu}_s^{(r_m)}\big) + \boldsymbol{\mu}_o^{(r_m)}, \tag{32}$$
$$\boldsymbol{A}^{(r_m)\text{-}1}\big(\boldsymbol{\Sigma}_s^{(m)} - \boldsymbol{\Sigma}_s^{(r_m)}\big)\boldsymbol{A}^{(r_m)\text{-}\mathsf{T}} + \boldsymbol{\Sigma}_o^{(r_m)}\big)$$

If $M = R$ then there is a one-to-one mapping of r_m to m, hence $\boldsymbol{\mu}_s^{(m)} = \boldsymbol{\mu}_s^{(r_m)}$, and $\boldsymbol{\Sigma}_s^{(m)} = \boldsymbol{\Sigma}_s^{(r_m)}$, making the differences between these terms zero, therefore from equation 32

$$p(\boldsymbol{o}_t|\theta_t, \mathcal{M}, \check{\mathcal{M}}, m) = \mathcal{N}\big(\boldsymbol{o}_t; \boldsymbol{\mu}_o^{(r_m)}, \boldsymbol{\Sigma}_o^{(r_m)}\big) \tag{33}$$
$$= \mathcal{N}\big(\boldsymbol{o}_t; \boldsymbol{\mu}_o^{(m)}, \boldsymbol{\Sigma}_o^{(m)}\big) \tag{34}$$

where $\boldsymbol{\mu}_o^{(r_m)}$ and $\boldsymbol{\Sigma}_o^{(r_m)}$ are derived from the clean speech model parameters using VTS, and hence equivalent to the VTS form. This result can be extrapolated to the `Joint` uncertainty method converging, when $R = M$, to whatever model compensation technique is used to derive the joint distribution needed to the compute the `Joint` parameters, including for example PMC or single-pass re-training [12].

This convergence of model-based `Joint` uncertainty decoding to whatever form of compensation is used to generate the joint distribution is a very useful property. It allows a flexibility in controlling the computational cost of the `Joint` scheme by adjusting the number of model classes R. Using a VTS approximation to compensate each acoustic model component, is equivalent to deriving a corrupted given the clean speech conditional for each component in equation 4; this involves computing and applying two Jacobian matrices for *each* component, both $\mathcal{O}(MD_s^2)$ costs, where recall D_s is the number of static features. In contrast, `Joint` transforms are estimated per *class*. This allows sharing the cost of computing the joint distribution over the class, which is far cheaper at $\mathcal{O}(RD_s^2)$, assuming the number of classes R is much smaller than the number of components M. The compensation itself may also be more efficient where only R means can be updated rather than all M components. The update of every model variance is also simpler with a single vector addition, rather than several matrix multiplies and an add necessary for VTS compensation.

The decoding form for model-based Joint uncertainty decoding is similar to CMLLR, where the features are transformed by a matrix and bias, however there is now an additional variance bias $\boldsymbol{\Sigma}_{\mathsf{b}}^{(r_m)}$. An efficient implementation of CMLLR uses multiple parallel features for each regression class to avoid changing the model parameters; however with `Joint` uncertainty decoding, the model variances and cached normalisation terms must be updated. However, whereas CMLLR transforms must be estimated from adaptation data, the joint distribution and hence `Joint` transforms may be predicted using prior models of the noise and clean speech. This implies the quality of the `Joint` transforms is solely dependent on the accuracy of the noise model and not on the number transforms/model classes. The noise model itself has relatively few parameters and may be estimated on very little data; in [21] only a few frames of noisy data is necessary to train an accurate model. With MLLR the amount of training data requires scales with the number of transforms used. Furthermore, if the noise is stationary it may be estimated in advance, in the background before the onset of speech, whereas CMLLR requires actual corrupted speech data.

3.3 Estimating `Joint` Compensation Parameters

In previous work with `Joint` uncertainty decoding [24, 25, 26], joint distributions were estimated using stereo data. While this provides an idealised method to evaluate this compensation form separate from the noise estimation process, this is not a requirement. A practical noise robustness technique should handle unseen noise conditions by automatically estimating the necessary compensation parameters given some un-transcribed data from the test environment. This section describes how the `Joint` uncertainty decoding form parameters may be derived given a prior model of the clean speech and a model of the corrupting noise.

The joint distribution, and hence `Joint` compensation parameters, may be estimated from the clean speech and a model of the noise, in a "predictive fashion" [15] by using VTS compensation to derive the necessary statistics of the joint distribution as mentioned in [24]. In the previous section it was shown how given a distribution of the clean speech and a noise model, the corrupted speech distribution can be estimated. For the `Joint` scheme, the cross-covariance is also necessary. This is derived by determining the covariance between the clean speech and the first-order VTS approximation, from equation 6, of the corrupted speech

$$\begin{aligned}\boldsymbol{\Sigma}_{xy}^{(r_m)} &= \mathcal{E}\left\{\boldsymbol{x}\boldsymbol{y}\right\} - \boldsymbol{\mu}_x\boldsymbol{\mu}_y && (35)\\ &\approx \mathcal{E}\left\{\boldsymbol{x}\boldsymbol{y}_{vts}^{\mathsf{T}}\right\} - \boldsymbol{\mu}_x\boldsymbol{\mu}_y^{\mathsf{T}} && (36)\end{aligned}$$

assuming independence between the clean speech and noise allows only the terms from the truncated series that affect the cross-covariance, i.e. are a function of the clean speech, to be considered

$$\begin{aligned}\boldsymbol{\Sigma}_{xy}^{(r_m)} &\approx \mathcal{E}\left\{\boldsymbol{x}\left[\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0}(\boldsymbol{x}-\boldsymbol{\mu}_x^{(r_m)})\right]^{\mathsf{T}}\right\} - \boldsymbol{\mu}_x^{(r_m)}\mathcal{E}\left\{\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0}(\boldsymbol{x}-\boldsymbol{\mu}_x^{(r_m)})\right\}^{\mathsf{T}} && (37)\\ &= \boldsymbol{\Sigma}_x^{(r_m)}\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0}^{\mathsf{T}} && (38)\end{aligned}$$

Similarly, the delta and delta-delta cross-covariances are

$$\begin{aligned}\boldsymbol{\Sigma}_{\Delta xy}^{(r_m)} &\approx \boldsymbol{\Sigma}_{\Delta x}^{(r_m)}\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0}^{\mathsf{T}} && (39)\\ \boldsymbol{\Sigma}_{\Delta^2 xy}^{(r_m)} &\approx \boldsymbol{\Sigma}_{\Delta^2 x}^{(r_m)}\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0}^{\mathsf{T}} && (40)\end{aligned}$$

Equations 38–40 provide the cross-covariance terms for equation 30. Hence the entire joint distribution may be computed from the clean speech parameters $\boldsymbol{\mu}_s^{r_m}$ and $\boldsymbol{\Sigma}_s^{r_m}$ and a noise model, using the derivations for the corrupted speech parameters from equations 9, 13, 20, 23, 24, and 25, and these cross-covariance terms. For example, the transform matrix in equation 29 is given

by

$$\boldsymbol{A}^{(r_m)} = \boldsymbol{\Sigma}_s^{(r_m)} \boldsymbol{\Sigma}_{os}^{(r_m)\text{-}1} \tag{41}$$

$$= \begin{bmatrix} \boldsymbol{\Sigma}_x^{(r_m)} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma}_{\Delta x}^{(r_m)} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \boldsymbol{\Sigma}_{\Delta^2 x}^{(r_m)} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Sigma}_{xy}^{(r_m)} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma}_{\Delta xy}^{(r_m)} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \boldsymbol{\Sigma}_{\Delta^2 xy}^{(r_m)} \end{bmatrix}^{\text{-T}} \tag{42}$$

$$\approx \begin{bmatrix} \boldsymbol{\Sigma}_x^{(r_m)} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma}_{\Delta x}^{(r_m)} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \boldsymbol{\Sigma}_{\Delta^2 x}^{(r_m)} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Sigma}_x^{(r_m)} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\big|_{\mu_0}^{\mathsf{T}} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma}_{\Delta x}^{(r_m)} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\big|_{\mu_0}^{\mathsf{T}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \boldsymbol{\Sigma}_{\Delta^2 x}^{(r_m)} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\big|_{\mu_0}^{\mathsf{T}} \end{bmatrix}^{\text{-T}} \tag{43}$$

This simplifies to

$$\boldsymbol{A}^{(r_m)} = \begin{bmatrix} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\big|_{\mu_0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\big|_{\mu_0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\big|_{\mu_0} \end{bmatrix}^{\text{-1}} \tag{44}$$

Recall that $\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\big|_{\mu_0} \rightarrow \boldsymbol{I}$ when the noise level is low: In this case, the transform is also identity indicating that no compensation is required. When the noise subsumes the speech $\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\big|_{\mu_0} \rightarrow \mathbf{0}$, which implies $\boldsymbol{A}^{(r_m)} \rightarrow \infty$. Some of the theoretical implications of this result for uncertainty decoding schemes are discussed in [27].

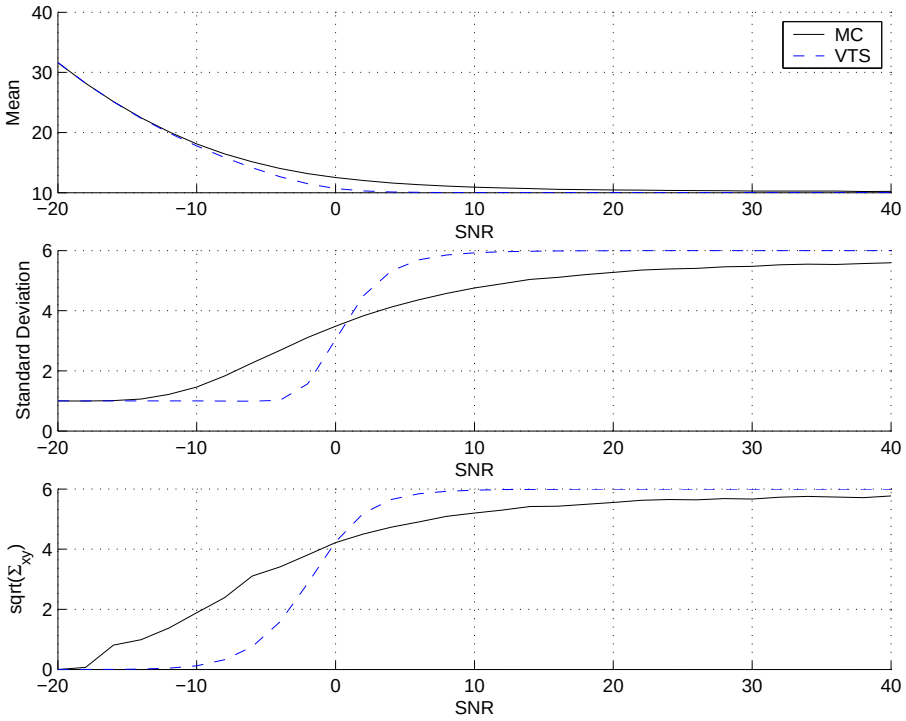

Figure 4: Comparing Monte Carlo and VTS generated corrupted speech distributions and cross-covariance.

The quality of using a VTS-based approximation for the corrupted speech may be investigated through a simulation in the log-spectral domain. Figure 4 shows how the first-order VTS approximation of the compensated mean and variances compare to a fitted distribution on the actual

Monte Carlo corrupted speech data where $y = \log(\exp(x) + \exp(n))$. The mean and standard deviation of the clean speech were 10 and 6 respectively, the variance of the noise 1, and the mean of the noise adjusted to achieve the SNR. The cross covariance between the clean and corrupted speech is also shown along with the first-order VTS estimated values. The VTS compensated mean appears quite accurate compared to the numerical result. The variance however is not as well approximated; this is similar to previous results [30, 1]. The cross-covariance between the clean and corrupted speech is also only roughly approximated.

Like CMLLR, `Joint` transforms compensate trained systems to more closely match the test environment; but `Joint` transforms have the addition of the variance bias. By using a VTS approximation to generate the joint distribution from models of the clean speech and noise, the associated `Joint` transform compensates for noise. However, the joint distribution is a general model of the clean $\boldsymbol{s}$ and observed $\boldsymbol{o}$ speech. Hence, the joint distribution may model other effects as well if this is taken into account during the generation of the joint distribution. For example, vocal tract length or a feature de-correlating transform could be incorporated in the mismatch function to generate a `Joint` transform that handles multiple factors. The joint distribution may also be thought of more generally as modeling training and test conditions jointly rather than as clean and noisy data. Hence `Joint` transforms may also be applied to multistyle systems to compensate for environmental mismatch.

3.3.1 The Clean Speech Class Model

The previous section demonstrated how the complete joint distribution for a model class r_m may be derived given a noise model and clean speech model; this clean speech model has not been defined. An a priori model of the clean speech $\mathcal{N}\big(\boldsymbol{\mu}_x^{(r_m)}, \boldsymbol{\Sigma}_x^{(r_m)}\big)$ is needed to determine the joint distribution per class r_m. There are three different approaches to deriving this Gaussian model. Diagonal covariances can be assumed throughout, however this may be a poor approximation since

$$\texttt{diag}\left\{\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0} \boldsymbol{\Sigma}_x^{(r_m)} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0}^{\mathsf{T}}\right\} \neq \texttt{diag}\left\{\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0} \boldsymbol{\Sigma}_{x,\texttt{diag}}^{(r_m)} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0}^{\mathsf{T}}\right\} \tag{45}$$

where recall the Jacobian matrix $\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}$ is full. Therefore, a full covariance form of the clean model is examined. There are two ways to estimate it. The first is to estimate a full covariance version of the acoustic model using the same alignments as the diagonal through single-pass re-training. The mean $\boldsymbol{\mu}_x^{(r_m)}$ and full variance $\boldsymbol{\Sigma}_x^{(r_m)}$ representing class r_m are then

$$L^{(m)} = \sum_{t=1}^{T} \gamma_t^{(m)} \tag{46}$$

$$L^{(r_m)} = \sum_{m \in r} L^{(m)} \tag{47}$$

$$\boldsymbol{\mu}_x^{(r_m)} = \frac{1}{L^{(r_m)}} \sum_{m \in r} L^{(m)} \boldsymbol{\mu}_x^{(r_m)} \tag{48}$$

$$\boldsymbol{\Sigma}_x^{(r_m)} = \frac{1}{L^{(r_m)}} \sum_{m \in r} L^{(m)} \left(\boldsymbol{\Sigma}_x^{(m)} + \boldsymbol{\mu}_x^{(m)} \boldsymbol{\mu}_x^{(m)^{\mathsf{T}}}\right) - \boldsymbol{\mu}_x^{(r_m)} \boldsymbol{\mu}_x^{(r_m)\mathsf{T}} \tag{49}$$

where $\boldsymbol{\mu}_x^{(m)}$ is the acoustic component mean, $\boldsymbol{\Sigma}_x^{(m)}$ the variance and $\gamma_t^{(m)}$ the component posterior/alignment probability at time frame t. An approximation to this is to use the diagonal form of $\boldsymbol{\Sigma}_x^{(m)}$. For low numbers of classes R compared to the number of model components M in the acoustic model, this should be a good approximation since the between class variance should dominate over the within class variance.

4 Noise Estimation Using EM

The previous section detailed two predictive forms of noise compensation: VTS-based and model-based `Joint`; these require a model of the noise environment. One option is to train the noise model from non-speech regions such as the first and last 10-30 frames of each utterance [35, 39]. This may provide a good model for short utterances, however some sentences may be sufficiently long that the noise environment changes while speech continues to be spoken as was found even on the AURORA2 [18] which is a short artificially corrupted digit string recognition task [35]. For example, in [41] the noise model is updated every 100 frames, however only those frames that are classified as non-speech are used; in this work, a noise estimation process which can estimate the noise even during speech is presented.

Moreno [30] first provided an Expectation-Maximisation (EM) framework to determine the means of the additive and convolutional noise in a ML fashion. Others have since successfully used this approach to estimate both convolutional and additive noise models [21, 10, 6, 40]; a similar approach is presented here, but in addition the additive noise variance is also concurrently estimated. This allows for *unsupervised* noise estimation of the full noise model whilst the speaker is still talking. Two forms are discussed that provide ML noise estimates for either VTS or `Joint` compensation. This overall task is not trivial since the solution space is multi-dimensional, highly complex and may have multiple local maxima. However, through iteration and assuring that successive estimates of the noise improve the likelihood of the training data, a reasonable model of the environment should be attained. Furthermore, noise models may be estimated for multistyle trained acoustic models; with the result being convolutional and additive noise estimates that best reduce the mismatch between the acoustic models and test conditions.

The objective of the noise estimation process is to estimate noise parameters $\hat{\mathbf{\Phi}} = \{\hat{\boldsymbol{\mu}}_n, \hat{\boldsymbol{\mu}}_h, \hat{\mathbf{\Sigma}}_n\}$, that given the acoustic model parameters $\mathcal{M}$, maximise the likelihood of the noisy observation sequence given by

$$\hat{\mathbf{\Phi}} = \arg\max_{\mathbf{\Phi}} p\left(\boldsymbol{O}|\mathcal{W}_{\texttt{hypo}}, \mathcal{M}, \mathbf{\Phi}\right) \tag{50}$$

A single Gaussian model of the additive noise is assumed; others have used a GMM to represent the additive noise such as [10], but the gains seem minimal compared to the computation cost [38]. The auxiliary, or $\mathcal{Q}$-function, for the expectation step of the EM algorithm is given by

$$\mathcal{Q}(\mathbf{\Phi}; \hat{\mathbf{\Phi}}) = \sum_{t=1}^{T} \sum_{m=1}^{M} \gamma_t^{(m)} \log \left[p(\boldsymbol{o}_t | \hat{\mathbf{\Phi}}, \mathcal{M}, m) \right] \tag{51}$$

where $\gamma_t^{(m)}$ is the state/component alignment probability at frame t, for the sentence hypothesis $\mathcal{W}_{\texttt{hypo}}$, given the acoustic model $\mathcal{M}$ and original noise model $\mathbf{\Phi}$. The form of output distribution $p(\boldsymbol{o}_t|\hat{\mathbf{\Phi}}, \mathcal{M}, m)$ differs depending on whether the noise estimates are to be optimised for VTS compensation or `Joint` uncertainty decoding. In either case, ML estimates of the static additive and convolutional noise means, and the static and dynamic additive noise variances should improve the auxiliary function given the same state alignment. That is find $\hat{\mathbf{\Phi}}$ such that

$$\mathcal{Q}(\mathbf{\Phi}; \hat{\mathbf{\Phi}}) \geq \mathcal{Q}(\mathbf{\Phi}; \mathbf{\Phi}) \tag{52}$$

This iterative EM-based ML formulation for estimating a model of the environment even during speech is shown in figure 5. With an initial estimate of the additive and convolutional noise: $\boldsymbol{\mu}_n$, $\mathbf{\Sigma}_n$ and $\boldsymbol{\mu}_h$, new estimates: $\hat{\boldsymbol{\mu}}_n$, $\hat{\mathbf{\Sigma}}_n$ and $\hat{\boldsymbol{\mu}}_h$, are computed. Either VTS or `Joint` compensation may be used in the auxiliary function to give ML noise estimates tailored for either form. The estimation may take place while speech is spoken, not only during non-speech regions allowing for noise model adaption during long speech segments. A single GMM may be used, to model the speech, for speed when computing the auxiliary function, such that decoding in the front-end is not required. However, in this work sufficient statistics are gathered using the actual full back-end acoustic models. The models may also be compensated with the initial noise parameters calculated in step 1, but in this work these initial estimates are only used during the estimation

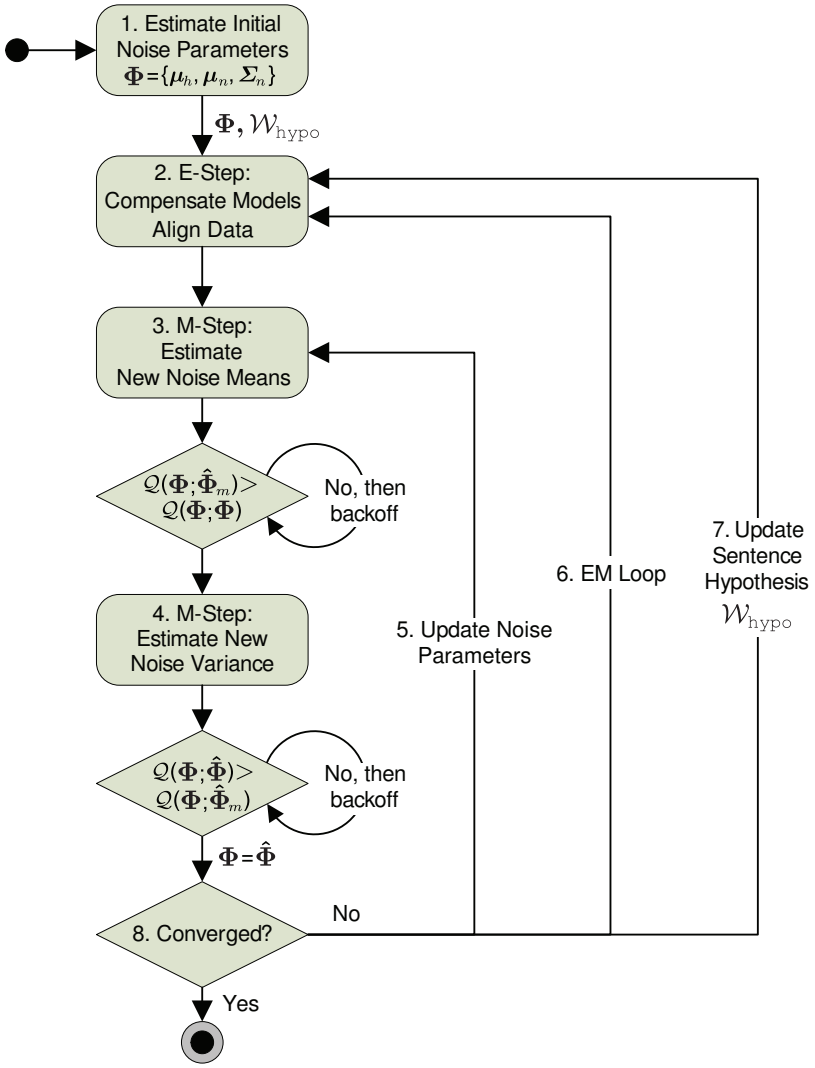

Figure 5: Noise parameter estimation procedure.

step. On successive EM iterations though, the updated noise parameters are used to compensate the models used during alignment.

4.1 Noise Estimation for VTS Compensation

An iterative solution for updating the static means of the additive and convolutional noise was given in [30]—this is modified to operate in the same domain many speech recognisers work in: the cepstral domain. With these new estimates of the noise means, the additive noise variance parameters are optimised using a simple gradient-based method. Together these give the updated noise parameter set $\hat{\boldsymbol{\Phi}}$.

For noise estimates to be optimised for VTS compensation, VTS compensation should be used to update the output distributions. Hence the logarithm of the probability, in equation 51 of the corrupted speech given the new model parameters, for a given state j and component m at time t, is

$$\begin{aligned}\log\left[p\Big(\boldsymbol{o}_t|\hat{\boldsymbol{\Phi}},\mathcal{M},m\Big)\right] = &-\frac{D}{2}\log(2\pi) \\ &-\frac{1}{2}\log|\hat{\boldsymbol{\Sigma}}_y^{(m)}| - \frac{1}{2}\Big(\boldsymbol{y}_t - \hat{\boldsymbol{\mu}}_y^{(m)}\Big)^{\mathsf{T}}\hat{\boldsymbol{\Sigma}}_y^{(m)\text{-}1}\Big(\boldsymbol{y}_t - \hat{\boldsymbol{\mu}}_y^{(m)}\Big) \\ &-\frac{1}{2}\log|\hat{\boldsymbol{\Sigma}}_{\Delta y}^{(m)}| - \frac{1}{2}\Big(\Delta\boldsymbol{y}_t - \boldsymbol{\mu}_{\Delta y}^{(m)}\Big)^{\mathsf{T}}\hat{\boldsymbol{\Sigma}}_{\Delta y}^{(m)\text{-}1}\Big(\Delta\boldsymbol{y}_t - \boldsymbol{\mu}_{\Delta y}^{(m)}\Big) \\ &-\frac{1}{2}\log|\hat{\boldsymbol{\Sigma}}_{\Delta^2 y}^{(m)}| - \frac{1}{2}\Big(\Delta^2\boldsymbol{y}_t - \boldsymbol{\mu}_{\Delta^2 y}^{(m)}\Big)^{\mathsf{T}}\hat{\boldsymbol{\Sigma}}_{\Delta^2 y}^{(m)\text{-}1}\Big(\Delta^2\boldsymbol{y}_t - \boldsymbol{\mu}_{\Delta^2 y}^{(m)}\Big)\end{aligned} \tag{53}$$

where D is the dimensionality of the feature vector and here equal to $3D_s$ as the sum of the number of static, velocity and acceleration coefficients. The corrupted speech parameters are derived from the clean using the VTS compensation form outlined previously in section 3.1.

4.1.1 Noise Parameter Initialisation

The first step (1) in the noise estimation process shown in figure 5 is to estimate initial noise parameters. Initialisation for the additive and convolutional noise means may be the minimum energy frame and the difference between the observed corrupted speech and the clean

$$\boldsymbol{\mu}_n = \min\{\boldsymbol{Y}\} \tag{54}$$

$$\boldsymbol{\mu}_h = \mathcal{E}\{\boldsymbol{Y}\} - \boldsymbol{\mu}_x \tag{55}$$

as in [30], where $\boldsymbol{\mu}_x$ is the global speech mean. For this work, the convolutional noise is instead initialised to zero.

The additive noise variance can be estimated from using the background noise frames [9, 39]. The initial value here of $\boldsymbol{\Sigma}_n$ is set to the variance of the first five frames of the observed speech. This is similar to the noise initialisation scheme in [20], where the first 3–4 frames are considered silence and used to initialise the noise model parameters. This should provide a much better estimate than using the global clean speech variance, which should be considered an upper bound, but may be worse than initialising it to zero if there is very little environmental noise [29]. In this work, if this first initialisation fails to provide a noise estimate that improves the auxiliary function, then the additive noise is set to $\boldsymbol{C}\boldsymbol{f}_0$ and zero variance, where $\boldsymbol{f}_0$ is the log zero vector, to represent a "no noise" condition.

4.1.2 Estimating the Static Noise Parameters

The maximisation of the static additive and convolutional noise means in step 3 is based on the ML formulation introduced in [30], but in the cepstral domain. The first-order Taylor series approximation in equation 6 may be used to express the corrupted speech mean as a function of

initial and new estimates additive and convolutional noise means

$$\hat{\mu}_{y,i}^{(m)} = \mathcal{E}\{y_i\} \tag{56}$$

$$\approx \mathcal{E}\left\{ y_i\big|_{\mu_0} + \boldsymbol{\nabla}_{\mathbf{x}}\, y_i\big|_{\mu_0} \cdot \left(\boldsymbol{x} - \boldsymbol{\mu}_x^{(m)}\right) + \boldsymbol{\nabla}_{\mathbf{n}}\, y_i\big|_{\mu_0} \cdot (\boldsymbol{n} - \boldsymbol{\mu}_n) + \boldsymbol{\nabla}_{\mathbf{h}}\, y_i\big|_{\mu_0} \cdot (\boldsymbol{h} - \boldsymbol{\mu}_h) \right\} \tag{57}$$

$$\approx \mu_{y,i}^{(m)} + \boldsymbol{\nabla}_{\mathbf{n}}\, y_i\big|_{\mu_0} \cdot (\hat{\boldsymbol{\mu}}_n - \boldsymbol{\mu}_n) + \boldsymbol{\nabla}_{\mathbf{h}}\, y_i\big|_{\mu_0} \cdot (\hat{\boldsymbol{\mu}}_h - \boldsymbol{\mu}_h) \tag{58}$$

assuming that the speech and noise are independent. This can also be written as

$$\hat{\boldsymbol{\mu}}_y^{(m)} = \boldsymbol{\mu}_y^{(m)} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} (\hat{\boldsymbol{\mu}}_n - \boldsymbol{\mu}_n) + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0} (\hat{\boldsymbol{\mu}}_h - \boldsymbol{\mu}_h) \tag{59}$$

recalling that $\big|_{\mu_0}$ indicates the function is evaluated at the Taylor series expansion point about the clean speech mean and the initial noise estimates. This form of the corrupted speech mean can be used to maximise the auxiliary function in equation 51 with respect to new noise mean estimates $\boldsymbol{\mu}_n$, $\boldsymbol{\mu}_h$ and a fixed additive noise variance.

To find new estimates of the additive and convolutional noise, we maximise the expected value of the auxiliary, differentiate it with respect to the delta parameters sought, equate to zero and solve. The partial derivatives of the auxiliary with respect to these parameters are

$$\frac{\partial}{\partial \hat{\boldsymbol{\mu}}_n} \mathcal{Q}(\boldsymbol{\Phi}; \hat{\boldsymbol{\Phi}}_{\hat{\mu}}) = \sum_{t=1}^{T} \sum_{m=1}^{M} \gamma_t^{(m)} \frac{\partial}{\partial \hat{\boldsymbol{\mu}}_n} \left[\log p(\boldsymbol{o}_t | \hat{\boldsymbol{\Phi}}_{\hat{\mu}}, \mathcal{M}, m) \right] \tag{60}$$

$$= \sum_{t=1}^{T} \sum_{m=1}^{M} \gamma_t^{(m)} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}^{\mathsf{T}} \boldsymbol{\Sigma}_y^{(m)\text{-}1} \times \tag{61}$$

$$\left(\boldsymbol{y}_t - \boldsymbol{\mu}_y^{(m)} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} \boldsymbol{\mu}_n + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0} \boldsymbol{\mu}_h - \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} \hat{\boldsymbol{\mu}}_n - \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0} \hat{\boldsymbol{\mu}}_h \right)$$

$$= \boldsymbol{d} - \boldsymbol{E}\hat{\boldsymbol{\mu}}_h - \boldsymbol{F}\hat{\boldsymbol{\mu}}_n \tag{62}$$

and

$$\frac{\partial}{\partial \hat{\boldsymbol{\mu}}_h} \mathcal{Q}(\boldsymbol{\Phi}; \hat{\boldsymbol{\Phi}}_{\hat{\mu}}) = \sum_{t=1}^{T} \sum_{m=1}^{M} \gamma_t^{(m)} \frac{\partial}{\partial \hat{\boldsymbol{\mu}}_h} \left[\log p(\boldsymbol{o}_t | \hat{\boldsymbol{\Phi}}_{\hat{\mu}}, \mathcal{M}, m) \right] \tag{63}$$

$$= \sum_{t=1}^{T} \sum_{m=1}^{M} \gamma_t^{(m)} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0}^{\mathsf{T}} \boldsymbol{\Sigma}_y^{(m)\text{-}1} \times \tag{64}$$

$$\left(\boldsymbol{y}_t - \boldsymbol{\mu}_y^{(m)} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} \boldsymbol{\mu}_n + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0} \boldsymbol{\mu}_h - \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} \hat{\boldsymbol{\mu}}_n - \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0} \hat{\boldsymbol{\mu}}_h \right)$$

$$= \boldsymbol{g} - \boldsymbol{H}\hat{\boldsymbol{\mu}}_h - \boldsymbol{J}\hat{\boldsymbol{\mu}}_n \tag{65}$$

A full derivation can be found in appendix B. Also, $\hat{\boldsymbol{\Phi}}_{\hat{\mu}} = \{\hat{\boldsymbol{\mu}}_n, \hat{\boldsymbol{\mu}}_h, \boldsymbol{\Sigma}_n\}$ which indicates that while these noise means are being updated, the additive noise variance is unchanged. These derivatives can be equated with zero to find the optimal points of the auxiliary function

$$\boldsymbol{g} - \boldsymbol{H}\hat{\boldsymbol{\mu}}_h - \boldsymbol{J}\hat{\boldsymbol{\mu}}_n = 0 \tag{66}$$

$$\boldsymbol{d} - \boldsymbol{E}\hat{\boldsymbol{\mu}}_h - \boldsymbol{F}\hat{\boldsymbol{\mu}}_n = 0 \tag{67}$$

which can be written in matrix form as

$$\begin{bmatrix} \boldsymbol{E} & \boldsymbol{F} \\ \boldsymbol{H} & \boldsymbol{J} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{\mu}}_h \\ \hat{\boldsymbol{\mu}}_n \end{bmatrix} = \begin{bmatrix} \boldsymbol{d} \\ \boldsymbol{g} \end{bmatrix} \tag{68}$$

where the accumulates $\boldsymbol{d}$, $\boldsymbol{g}$, $\boldsymbol{E}$, $\boldsymbol{F}$, $\boldsymbol{H}$, and $\boldsymbol{J}$ are defined as

$$\begin{aligned}
\boldsymbol{d} &= \sum_{m=1}^{M} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}^{\mathsf{T}} \boldsymbol{\Sigma}_y^{(m)\text{-}1} \sum_{t=1}^{T} \gamma_t^{(m)} \left(\boldsymbol{y}_t - \boldsymbol{\mu}_y^{(m)} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} \boldsymbol{\mu}_n + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0} \boldsymbol{\mu}_h \right) \\
\boldsymbol{g} &= \sum_{m=1}^{M} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0}^{\mathsf{T}} \boldsymbol{\Sigma}_y^{(m)\text{-}1} \sum_{t=1}^{T} \gamma_t^{(m)} \left(\boldsymbol{y}_t - \boldsymbol{\mu}_y^{(m)} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} \boldsymbol{\mu}_n + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0} \boldsymbol{\mu}_h \right) \\
\boldsymbol{E} &= \sum_{m=1}^{M} L^{(m)} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}^{\mathsf{T}} \boldsymbol{\Sigma}_y^{(m)\text{-}1} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0} \qquad \boldsymbol{F} = \sum_{m=1}^{M} L^{(m)} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}^{\mathsf{T}} \boldsymbol{\Sigma}_y^{(m)\text{-}1} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} \\
\boldsymbol{H} &= \sum_{m=1}^{M} L^{(m)} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0}^{\mathsf{T}} \boldsymbol{\Sigma}_y^{(m)\text{-}1} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0} \qquad \boldsymbol{J} = \sum_{m=1}^{M} L^{(m)} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0}^{\mathsf{T}} \boldsymbol{\Sigma}_y^{(m)\text{-}1} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}
\end{aligned} \tag{69}$$

recalling $L^{(m)} = \sum_{t=1}^{T} \gamma_t^{(m)}$. Note that $\boldsymbol{E} = \boldsymbol{J}^{\mathsf{T}}$. Solving the linear system in equation 68 gives the following formulae for the parameters to be estimated

$$\hat{\boldsymbol{\mu}}_h = (\boldsymbol{H} - \boldsymbol{J}\boldsymbol{F}^{\text{-}1}\boldsymbol{E})^{\text{-}1}(\boldsymbol{g} - \boldsymbol{J}\boldsymbol{F}^{\text{-}1}\boldsymbol{d}) \tag{70}$$

$$\begin{aligned}
\hat{\boldsymbol{\mu}}_n &= (\boldsymbol{J} - \boldsymbol{H}\boldsymbol{E}^{\text{-}1}\boldsymbol{F})^{\text{-}1}(\boldsymbol{g} - \boldsymbol{H}\boldsymbol{E}^{\text{-}1}\boldsymbol{d}) \\
&= (\boldsymbol{F} - \boldsymbol{E}\boldsymbol{H}^{\text{-}1}\boldsymbol{J})^{\text{-}1}(\boldsymbol{d} - \boldsymbol{E}\boldsymbol{H}^{\text{-}1}\boldsymbol{g})
\end{aligned} \tag{71}$$

These can be used as the starting estimates for another iteration of EM, to determine new noise estimates until the auxiliary function fails to increase indicating convergence.

4.1.3 Backing Off

An issue arises with this estimation procedure. Until the new noise mean estimates $\hat{\boldsymbol{\Phi}}_{\hat{\mu}}$ converge and match the noise parameters used for the expansion point $\boldsymbol{\mu}_0$, the compensation used for the log likelihood calculation does not match the compensation used for the auxiliary function. That is

$$\boldsymbol{\mu}_y^{(m)} \neq \hat{\boldsymbol{\mu}}_y^{(m)} \tag{72}$$

The estimation procedure described ensures

$$\mathcal{Q}(\boldsymbol{\Phi}; \boldsymbol{\mu}_0 = \boldsymbol{\Phi}, \hat{\boldsymbol{\Phi}}_{\hat{\mu}}) \geq \mathcal{Q}(\boldsymbol{\Phi}; \boldsymbol{\mu}_0 = \boldsymbol{\Phi}, \boldsymbol{\Phi}) \tag{73}$$

but not that

$$\mathcal{Q}(\boldsymbol{\Phi}; \boldsymbol{\mu}_0 = \hat{\boldsymbol{\Phi}}_{\hat{\mu}}, \hat{\boldsymbol{\Phi}}_{\hat{\mu}}) \geq \mathcal{Q}(\boldsymbol{\Phi}; \boldsymbol{\mu}_0 = \boldsymbol{\Phi}, \boldsymbol{\Phi}) \tag{74}$$

where the additional term $\boldsymbol{\mu}_0 = \boldsymbol{\Phi}$ indicates the noise parameters that are used in the expansion point of the VTS approximation in the component output probability of equation 53. It is the latter equation 74 that must be true for the overall EM algorithm to converge. Hence, the new estimate needs to be backed off towards the old until this requirement is met.

The back-off strategy used increases the scalar parameter η from zero, such that for $\hat{\boldsymbol{\Phi}}_{\hat{\mu}\texttt{backoff}}$

$$\mathcal{Q}(\boldsymbol{\Phi}; \boldsymbol{\mu}_0 = \hat{\boldsymbol{\Phi}}_{\hat{\mu}\texttt{backoff}}, \hat{\boldsymbol{\Phi}}_{\hat{\mu}\texttt{backoff}}) \geq \mathcal{Q}(\boldsymbol{\Phi}; \boldsymbol{\mu}_0 = \boldsymbol{\Phi}, \boldsymbol{\Phi}) \tag{75}$$

where

$$\hat{\boldsymbol{\Phi}}_{\hat{\mu}\texttt{backoff}} = \eta \cdot \boldsymbol{\Phi} + (1 - \eta) \cdot \hat{\boldsymbol{\Phi}}_{\hat{\mu}} \tag{76}$$

If the auxiliary function again does not improve over $\mathcal{Q}(\boldsymbol{\Phi}; \boldsymbol{\mu}_0 = \boldsymbol{\Phi}, \boldsymbol{\Phi})$, then $\eta = \eta^2$. This is repeated until the auxiliary function is no longer worse than the original value, or a floor on η is reached. Setting η to a half produced reasonable results. To speed up the backing off process, an approximation may be made whereby only the static feature portion of the output probability in equation 53 is considered when computing the auxiliary. In this work, this is used as a first step for backing off, however after an increase in the auxiliary function for just the static features is found, the full auxiliary function is subsequently tested for improvement.

4.1.4 Estimating the Additive Noise Variance

A first-order gradient ascent method is used here to optimise the additive noise variance in the next maximisation step

$$\frac{\partial}{\partial \hat{\boldsymbol{\Sigma}}_n} \mathcal{Q}(\boldsymbol{\Phi}; \hat{\boldsymbol{\Phi}}) = \sum_{t=1}^{T} \sum_{m=1}^{M} \gamma_t^{(m)} \frac{\partial}{\partial \hat{\boldsymbol{\Sigma}}_n} \Big[\log p\Big(\boldsymbol{o}_t \big| \hat{\boldsymbol{\Phi}}, \mathcal{M}, m \Big) \Big] \tag{77}$$

$$= \sum_{t=1}^{T} \sum_{m=1}^{M} \gamma_t^{(m)} \frac{\partial}{\partial \hat{\boldsymbol{\Sigma}}_n} \left[-\frac{1}{2} \log |\hat{\boldsymbol{\Sigma}}_y^{(m)}| - \frac{1}{2} \Big(\boldsymbol{y}_t - \boldsymbol{\mu}_y^{(m)} \Big)^{\mathsf{T}} \hat{\boldsymbol{\Sigma}}_y^{(m)\text{-}1} \Big(\boldsymbol{y}_t - \boldsymbol{\mu}_y^{(m)} \Big) \right] \tag{78}$$

The gradient of the auxiliary function with respect to the static noise variances from equation 78 simplifies to

$$\frac{\partial}{\partial \hat{\sigma}_{n,i}^2} \mathcal{Q}(\boldsymbol{\Phi}; \hat{\boldsymbol{\Phi}}) = -\frac{1}{2} \sum_{m=1}^{M} \sum_{d=1}^{D_s} \left(\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}} \Big|_{\mu_0} \right)_{d,i}^2 \frac{1}{\hat{\sigma}_{y,d}^{(m)2}} \left\{ \left(1 - \frac{\mu_{y,d}^{(m)2}}{\hat{\sigma}_{y,d}^{(m)2}} \right) L^{(m)} - \frac{p_d^{(m)} - 2 q_d^{(m)} \mu_{y,d}^{(m)}}{\hat{\sigma}_{y,d}^{(m)2}} \right\} \tag{79}$$

where the sufficient statistics $\boldsymbol{p}^{(m)}$ and $\boldsymbol{q}^{(m)}$ are defined as

$$p_d^{(m)} = \sum_{t=1}^{T} \gamma_t^{(m)} y_{t,d}^2 \qquad\qquad q_d^{(m)} = \sum_{t=1}^{T} \gamma_t^{(m)} y_{t,d} \tag{80}$$

Refer to appendix C for a full derivation. There is no closed-form solution to find $\hat{\sigma}_{n,i}^2$ that zeros equation 79; there is no obvious way to re-express these gradients such that the noise variance can be factored out of the triple summation. Hence, to evaluate the gradient for different noise variances, the summation must always be performed, slowing any possible gradient ascent techniques. Contrast this with the auxiliary gradients with respect to the noise means; these can be computed directly from the accumulates $\boldsymbol{d}$, $\boldsymbol{E}$, $\boldsymbol{F}$, $\boldsymbol{g}$, $\boldsymbol{H}$, and $\boldsymbol{J}$. The additive noise variance update formula for the static noise variance is then

$$\hat{\boldsymbol{\Sigma}}_n = \boldsymbol{\Sigma}_n + \nu \frac{\partial}{\partial \hat{\boldsymbol{\Sigma}}_n} \mathcal{Q}(\boldsymbol{\Phi}; \hat{\boldsymbol{\Phi}}) \tag{81}$$

where ν is a scalar learning rate and in this work set to unity.

There is no guarantee that the step taken will improve the auxiliary—the step may be too large and significantly overshoot it. Hence, it is important to also back off the new variance estimate towards the old, as with the means in section 4.1.3. Also, as this is a gradient based maximisation process, the estimate of the variance may be refined with multiple iterations until a certain number of iterations has passed, or the auxiliary function fails to increase beyond a threshold. In this work, a maximum number of iterations is set, 10 was sufficient. The iterative nature of this maximisation step is not reflected in figure 5.

4.1.5 Additive Noise Dynamic Variances Estimation

If the noise is assumed to be fairly stationary, then the additive noise dynamic variances may be computed or predicted directly from the static noise variance. The delta coefficients are calculated in the following manner

$$\Delta \boldsymbol{y}_t = \frac{\sum_{\delta=1}^{\Delta} \delta (\boldsymbol{y}_{t+\delta} - \boldsymbol{y}_{t-\delta})}{2 \sum_{\delta=1}^{\Delta} \delta^2} \tag{82}$$

Hence for the delta variance of the noise

$$\boldsymbol{\Sigma}_{\Delta n} = \mathtt{Var} \left\{ \frac{\sum_{\delta=1}^{\Delta} \delta (\boldsymbol{n}_{t+\delta} - \boldsymbol{n}_{t-\delta})}{2 \sum_{\delta=1}^{\Delta} \delta^2} \right\} \tag{83}$$

$$= \sum_{\delta=1}^{\Delta} 2 \left(\frac{\delta}{2 \sum_{\delta=1}^{\Delta} \delta^2} \right)^2 \mathtt{Var} \{ \boldsymbol{n}_t \} \tag{84}$$

assuming independence between $\boldsymbol{n}_{t+\delta}$ and $\boldsymbol{n}_{t-\delta}$. For the window size used in this work of $\Delta = 2$, this simplifies to

$$\boldsymbol{\Sigma}_{\Delta n} = \frac{1}{10}\boldsymbol{\Sigma}_n \tag{85}$$

The delta-delta parameters can be similarly derived such that

$$\boldsymbol{\Sigma}_{\Delta^2 n} = \frac{1}{10}\boldsymbol{\Sigma}_{\Delta n} = \frac{1}{100}\boldsymbol{\Sigma}_n \tag{86}$$

for the same window size.

Using these forms for the dynamic additive noise variance tended to underestimate the ML value especially for the acceleration terms. Therefore, a gradient-based optimisation routine was used to find the ML estimates of the dynamic noise variances. The auxiliary gradients for dynamic coefficients of the noise variance are given by

$$\begin{aligned}
\frac{\partial \mathcal{Q}(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}})}{\partial \hat{\sigma}^2_{\Delta n,i}} &= -\frac{1}{2}\sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_t^{(m)}\frac{\partial}{\partial \hat{\sigma}^2_{\Delta n,i}}\left[\log|\hat{\boldsymbol{\Sigma}}^{(m)}_{\Delta y}| + \left(\Delta \boldsymbol{y}_t - \boldsymbol{\mu}^{(m)}_{\Delta y}\right)^{\mathsf{T}}\hat{\boldsymbol{\Sigma}}^{(m)\text{-}1}_{\Delta y}\left(\Delta \boldsymbol{y}_t - \boldsymbol{\mu}^{(m)}_{\Delta y}\right)\right] \\
&= -\frac{1}{2}\sum_{m=1}^{M}\sum_{d=1}^{D_s}\left(\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}\right)^2_{d,i}\frac{1}{\hat{\sigma}^{(m)2}_{\Delta y,d}}\left\{\left(1 - \frac{\mu^{(m)2}_{\Delta y,d}}{\hat{\sigma}^{(m)2}_{\Delta y,d}}\right)L^{(m)} - \frac{p^{(m)}_{\Delta d} - 2q^{(m)}_{\Delta d}\mu^{(m)}_{\Delta y,d}}{\hat{\sigma}^{(m)2}_{\Delta y,d}}\right\} \qquad (87) \\
\frac{\partial \mathcal{Q}(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}})}{\partial \hat{\sigma}^2_{\Delta^2 n,i}} &= -\frac{1}{2}\sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_t^{(m)}\frac{\partial}{\partial \hat{\sigma}^2_{\Delta^2 n,i}}\left[\log|\hat{\boldsymbol{\Sigma}}^{(m)}_{\Delta^2 y}| + \left(\Delta^2 \boldsymbol{y}_t - \boldsymbol{\mu}^{(m)}_{\Delta^2 y}\right)^{\mathsf{T}}\hat{\boldsymbol{\Sigma}}^{(m)\text{-}1}_{\Delta^2 y}\left(\Delta^2 \boldsymbol{y}_t - \boldsymbol{\mu}^{(m)}_{\Delta^2 y}\right)\right] \\
&= -\frac{1}{2}\sum_{m=1}^{M}\sum_{d=1}^{D_s}\left(\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}\right)^2_{d,i}\frac{1}{\hat{\sigma}^{(m)2}_{\Delta^2 y,d}}\left\{\left(1 - \frac{\mu^{(m)2}_{\Delta^2 y,d}}{\hat{\sigma}^{(m)2}_{\Delta^2 y,d}}\right)L^{(m)} - \frac{p^{(m)}_{\Delta^2 d} - 2q^{(m)}_{\Delta^2 d}\mu^{(m)}_{\Delta^2 y,d}}{\hat{\sigma}^{(m)2}_{\Delta^2 y,d}}\right\} \qquad (88)
\end{aligned}$$

which are similar to the static variance gradient since the dynamic variances are treated in a completely parallel manner. That is the derivatives of the determinant and main probabilities term are simplified in the same way, to the statics and

$$\begin{aligned}
\frac{\partial \hat{\boldsymbol{\Sigma}}^{(m)}_{y}}{\partial \hat{\sigma}^2_{n,i}} &= \frac{\partial \hat{\boldsymbol{\Sigma}}^{(m)}_{\Delta y}}{\partial \hat{\sigma}^2_{\Delta n,i}} = \frac{\partial \hat{\boldsymbol{\Sigma}}^{(m)}_{\Delta^2 y}}{\partial \hat{\sigma}^2_{\Delta^2 n,i}} \qquad (89) \\
&= \left(\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|^{\mathsf{T}}_{\mu_0}\right)_i\left(\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}\right)_i \qquad (90)
\end{aligned}$$

as can be observed from equation 173 found in appendix C. These terms are then used in the full static noise variance update formula

$$\begin{bmatrix} \hat{\boldsymbol{\Sigma}}_n \\ \hat{\boldsymbol{\Sigma}}_{\Delta n} \\ \hat{\boldsymbol{\Sigma}}_{\Delta^2 n} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Sigma}_n \\ \boldsymbol{\Sigma}_{\Delta n} \\ \boldsymbol{\Sigma}_{\Delta^2 n} \end{bmatrix} + \nu \begin{bmatrix} \frac{\partial}{\partial \hat{\boldsymbol{\Sigma}}_n}\mathcal{Q}(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}) \\ \frac{\partial}{\partial \hat{\boldsymbol{\Sigma}}_{\Delta n}}\mathcal{Q}(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}) \\ \frac{\partial}{\partial \hat{\boldsymbol{\Sigma}}_{\Delta^2 n}}\mathcal{Q}(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}) \end{bmatrix} \tag{91}$$

Like the noise means update, the auxiliary function should be evaluated with the new additive noise variance estimate, and if there in no improvement, the estimate should be backed off until there is, the previous auxiliary value is reached, or η reaches some minimum value.

4.1.6 Iterative Noise Estimation

There are many loops that can occur during this noise estimation process. The first, mentioned in the previous section and not shown in figure 5 is the iterative gradient ascent procedure in the second maximisation step. The next may be a complete iteration over the maximisation step

of both the noise means and variances, labeled step 5. This updates the VTS expansion point, recomputing the necessary terms for updating the noise parameters with the new set of noise parameters, using the sufficient statistics already accumulated, all crucially without going over the entire dataset again. That is $\gamma_t^{(m)}$, $L^{(m)}$, $\boldsymbol{p}_m$ and $\boldsymbol{q}_m$ remain unchanged over this loop. With the EM loop, labeled step 6, the expectation step is taken again, the acoustic model is compensated with the new noise parameter estimates, and these sufficient statistics are re-accumulated. Hence step 5 can be conducted theoretically far faster than step 6. The last loop updates the sentence hypothesis, using the updated noise parameters to compensate the acoustic models when re-decoding the test sentence; this of course is a rather expensive step compared to the previous two. In this work, it was found that iterating over the maximisation step with a successively improving VTS expansion point and noise variance, and only one EM step and the initial noise hypothesis was an effective strategy. Conducting another EM step, and improving the hypothesis, however do improve the noise estimates.

4.2 Noise Estimation for `Joint` Compensation

The previous section described a noise estimation procedure directed at obtaining an ML noise model presuming a VTS compensation scheme. While the `Joint` form may perform sufficiently well with such noise parameters, they are not optimal in an ML sense. Although the `Joint` form converges to VTS compensation when the number of classes equals the number of model components, when this is not the case the auxiliary functions are different. Hence, if the noise estimates are to be used in `Joint` uncertainty compensation, then an alternate auxiliary function during noise estimation should be used. This section outlines differences between the VTS compensation described previously, and `Joint` noise parameter estimation.

The same form of auxiliary from equation 51 is used for `Joint` noise estimation

$$\mathcal{Q}_J(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}) = \sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_t^{(m)}\log\Big[p_J(\boldsymbol{o}_t|\hat{\boldsymbol{\Phi}},\mathcal{M},m)\Big] \tag{92}$$

except the log probability for the output distribution is now given by

$$\log p_J(\boldsymbol{o}_t|\hat{\boldsymbol{\Phi}},\mathcal{M},m) = \log\Big[|\hat{\boldsymbol{A}}^{(r_m)}|\mathcal{N}\left(\hat{\boldsymbol{A}}^{(r_m)}\boldsymbol{o}_t + \hat{\boldsymbol{b}}^{(r_m)};\boldsymbol{\mu}_s^{(m)},\boldsymbol{\Sigma}_s^{(m)} + \hat{\boldsymbol{\Sigma}}_{\mathtt{b}}^{(r_m)}\right)\Big] \tag{93}$$

The full set of `Joint` transforms $\mathcal{T} = \left[\mathcal{T}^{(1)},\mathcal{T}^{(r_m)},\ldots,\mathcal{T}^{(R)}\right]$ may be derived from the joint distribution that is estimated from the clean speech class model and the estimated noise parameters $\hat{\boldsymbol{\Phi}}$.

4.2.1 Noise Parameter Initialisation

The various initialisation possibilities described for VTS based noise estimation could be applied here. However, it would be advantageous to first estimate the noise environment using the VTS form described earlier, and then refine the estimates for `Joint` compensation using the method described in this section. In practice, this was found to yield better results than using noise parameter initialisation as described in 4.1.1. However, it is necessary to check if the resulting noise model improves the auxiliary as in some cases, with low amounts of noise, it was better to start from a "no noise" parameter initialisation.

4.2.2 Estimating Noise Parameters for `Joint` Compensation

Given the acoustic model $\mathcal{M}$, from which the clean speech class model may be derived, an estimate of the noise parameters $\hat{\boldsymbol{\Phi}}$ that maximises the auxiliary function $\mathcal{Q}_J$ is required. That is find

$$\hat{\boldsymbol{\Phi}} = \Big\{\hat{\boldsymbol{\mu}}_n,\hat{\boldsymbol{\Sigma}}_n,\hat{\boldsymbol{\mu}}_h\Big\} = \arg\max_{\hat{\boldsymbol{\Phi}}}\mathcal{Q}_J\Big(\mathcal{M},\mathcal{T};\mathcal{M},\hat{\mathcal{T}}\Big) \tag{94}$$

where $\hat{\mathcal{T}}$ is computed directly from clean speech class model and $\hat{\mathbf{\Phi}}$. Given a suitable initial starting point, the noise parameters may be iteratively updated using a simple gradient-based optimisation scheme

$$\hat{\mu}_{n,i} = \mu_{n,i} - \zeta \frac{\frac{\partial \mathcal{Q}_J}{\partial \mu_{n,i}}}{\frac{\partial^2 \mathcal{Q}_J}{\partial \mu_{n,i}^2}} \tag{95}$$

$$\hat{\sigma}^2_{n,i} = \sigma^2_{n,i} - \zeta \frac{\frac{\partial \mathcal{Q}_J}{\partial \sigma^2_{n,i}}}{\frac{\partial^2 \mathcal{Q}_J}{\partial (\sigma^2_{n,i})^2}} \tag{96}$$

$$\hat{\mu}_{h,i} = \mu_{h,i} - \zeta \frac{\frac{\partial \mathcal{Q}_J}{\partial \mu_{h,i}}}{\frac{\partial^2 \mathcal{Q}_J}{\partial \mu_{h,i}^2}} \tag{97}$$

where ζ is the learning rate.The second derivatives need to be conditioned such that they remain negative to ensure the updates converges to a local maximum; in case they are not negative, a simple back-off strategy is to use a fixed step size with the first-order gradient. It is also important to ensure that each step in the iteration improves the auxiliary and hence a multi-tiered back-off of the estimates generated is used similar to the VTS noise mean back-off strategy outlined in section 4.1.3. First the statics are interpolated between the new and old until the auxiliary function has increased, then the velocity coefficients and finally accelerations; the means and variances are estimated separately. For this work, numerical derivatives of the `Joint` auxiliary function are used.

5 `Joint` Adaptive Training

Adaptation has been shown to be a powerful technique to reduce the acoustic mismatch between training and test conditions [14, 23, 31]. When there is insufficient data to retrain models to match the testing condition, adaptation provides a efficient way to include data, if any, from the test condition. While adaptation of well trained models has shown to be quite effective, even more powerful is the adaptive training technique first described in [2]. Here, transforms per speaker are applied during training to yield a canonical speaker independent acoustic model. During testing, another transform trained on minimal test data, applied to the canonical model, factors in the speaker. In this case, adaptive training is described for speaker adaptation, but it can be more generally applied to reduce the acoustic mismatch from many factors such as the speaker, channel and environmental variability [16]. Using adaptive training yields a "purer" acoustic model than multi-style techniques that need to incorporate all the extraneous variability due to non-speech factors in the models. Moreover, the resulting canonical model may be a better "clean" acoustic model that all predictive noise compensation techniques require.

Linear transforms like MLLR [2] and CMLLR [14] have been successfully used in adaptive training. In this work, the use of `Joint` uncertainty transforms in an adaptive training framework is explored. Rather than separately modeling the speaker and noise condition with a MLLR transform and cluster adaptive training respectively as in [16], a `Joint` transform will model both for each speaker/noise condition. Applying `Joint` transforms in an adaptive training framework should better remove the noise in the data as they more effectively model the effects of noise on speech than CMLLR or a simple cepstral mean bias term. The uncertainty variance bias term allows JAT to train a "cleaner" canonical acoustic model by de-weighting noisy observations.

5.1 The Adaptive Training Framework

Adaptive training factors out non-linguistic speech variability such as different speakers and acoustic environments through the application of transforms during training and testing to a pure canonical model. Thus there are two distinct sets of parameters that must be estimated in a adaptive training framework:

- **Canonical model**
  This represents only the variability of the speech due to the different realisation of different sounds required to distinguish one phone from another. Typically this is an HMM, and represented by $\mathcal{M}$. Cluster based adaptive training with sets of HMMs forming the canonical model is not be considered here.

- **Transforms**
  These represent the non-linguistic speech acoustic variability due to different speakers or environments and are applied to the features or canonical model parameters to adapt the system to each specific homogeneous condition. The entire set of transforms is denoted by $\mathcal{T}$.

The parameter sets $\mathcal{M}$ and $\mathcal{T}$ are estimated such that they maximise the likelihood of the training data

$$p\left(\boldsymbol{O}|\mathcal{W}_{\texttt{trans}},\mathcal{M},\mathcal{T}\right)=\sum_{h=1}^{H}\sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_t^{(m)}p(\boldsymbol{o}_t|\mathcal{M},\mathcal{T},m) \tag{98}$$

where $\gamma_t^{(m)}$ is now the posterior probability that the observation $\boldsymbol{o}_t$ is generated by component m, transformed by `Joint` transform $\mathcal{T}^{(r_m h)}$, on heterogeneous training data segmented into H homogeneous blocks. The posterior probabilities should be computed over all possible sequences for each block h of length $T^{(h)}$, given the transcription $\mathcal{W}_{\texttt{trans}}^{(h)}$, and model set $\mathcal{M}$. The class index r_m, which associates the transform to the model component, was described previously in section 3.2. Again, EM is used to iteratively find suitable canonical model parameters and the noise

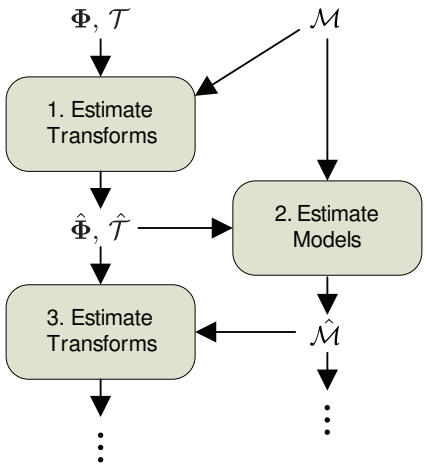

Figure 6: JAT—updating the clean speech class model.

parameters to generate the `Joint` transforms. The `Joint` auxiliary function, from equation 92 and used to give ML estimates of the noise for `Joint` compensation, may be extended to

$$\mathcal{Q}_J\left(\mathcal{M},\mathcal{T};\hat{\mathcal{M}},\hat{\mathcal{T}}\right)=\sum_{h=1}^{H}\sum_{t=1}^{T^{(h)}}\sum_{m=1}^{M}\gamma_t^{(m)}\log\left[|\hat{\boldsymbol{A}}^{(r_m h)}|\mathcal{N}\left(\hat{\boldsymbol{A}}^{(r_m h)}\boldsymbol{o}_t+\hat{\boldsymbol{b}}^{(r_m h)};\hat{\boldsymbol{\mu}}_s^{(m)},\hat{\boldsymbol{\Sigma}}_s^{(m)}+\hat{\boldsymbol{\Sigma}}_{\mathsf{b}}^{(r_m h)}\right)\right] \tag{99}$$

$$=\sum_{h=1}^{H}\sum_{t=1}^{T^{(h)}}\sum_{m=1}^{M}\gamma_t^{(m)}\log\mathcal{N}\left(\boldsymbol{o}_t;\hat{\boldsymbol{A}}^{(r_m h)\text{-}1}\left(\hat{\boldsymbol{\mu}}_s^{(m)}-\hat{\boldsymbol{\mu}}_s^{(r_m)}\right)+\hat{\boldsymbol{\mu}}_y^{(r_m h)},\right. \tag{100}$$
$$\left.\hat{\boldsymbol{A}}^{(r_m h)\text{-}1}\left(\hat{\boldsymbol{\Sigma}}_s^{(m)}-\hat{\boldsymbol{\Sigma}}_s^{(r_m)}\right)\hat{\boldsymbol{A}}^{(r_m h)\text{-}\mathsf{T}}+\hat{\boldsymbol{\Sigma}}_y^{(r_m h)}\right)$$

In adaptive training, both a set of transforms, and the acoustic model parameters are iteratively estimated in a generalised EM framework. Figure 6 shows one and a half iterations of `Joint` interleaved `Joint` adaptive training. The symbol $\boldsymbol{\Phi}$, for example, represents the noise parameters for iteration 1 that are associated with transform $\mathcal{T}$. First, given the current acoustic models $\mathcal{M}$ a new set of transform $\mathcal{T}$ is estimated. Subsequently, the canonical model parameters are updated to $\hat{\mathcal{M}}$ given this new set of transforms. Multiple iterations of this interleaved training may be performed to optimise the auxiliary function. The overall training regime may be summarised as

1. Initialise canonical model and transform parameters. These can be a well trained HMM and identity values respectively.
2. Estimate `Joint` transforms $\hat{\mathcal{T}}$, given $\mathcal{T}$ and $\mathcal{M}$.
3. Estimate the canonical model parameters $\hat{\mathcal{M}}$, given $\hat{\mathcal{T}}$ and $\mathcal{M}$.
4. Continue back to 2 until convergence.

By applying each new estimate is used in the maximisation of the next set of parameters, this ensures that the likelihood of the training data will increase by using the new set of all parameters.

5.2 Estimating the `Joint` Transform

In section 4.2, noise estimation based on finding a ML noise model given a `Joint` compensation auxiliary function was discussed; in the `Joint` adaptive training framework, transforms are estimated from this noise model and the clean speech class model. This clean speech class model,

described in 3.3.1, needs to be re-computed every time the canonical model is updated. A disconnect may arise when during the estimation of a new set of transforms, the initial ML noise parameters may have been estimated using a different clean speech class model. This problem can be clearly understood by following the adaptive training process in figure 6. The set of optimal transforms $\hat{\mathcal{T}}$ for the training data is computed from $\hat{\mathbf{\Phi}}$ and the clean speech class model derived from $\mathcal{M}$ and used in step 2, where a new set of canonical model parameters $\hat{\mathcal{M}}$ are estimated. But when step 3 starts, during the expectation step, the set of transforms generated from $\hat{\mathbf{\Phi}}$ and clean speech class model from $\hat{\mathcal{M}}$ is not the same as $\hat{\mathcal{T}}$, which is the set of transforms that EM requires to be the initial starting point.

Nevertheless, it may be possible to begin with the `Joint` transform produced from $\hat{\mathbf{\Phi}}$ and $\hat{\mathcal{M}}$. However, not only is it now necessary to verify the newly estimated `Joint` transform yields a higher auxiliary function value than the initial parameters, but that it also exceeds the auxiliary function value using the input joint transforms, in this example $\mathcal{T}_1$, which were computed from the previous clean speech class model and the initial noise parameters. It may be the case that due to the change in clean speech class model, that the newly estimated parameters may not improve the auxiliary function over the input transform, which was computed from a different clean speech class model. Currently, there is no convenient way to address this issue.

5.3 Estimating the Canonical Model Parameters

After a new set of transforms are estimated, the model parameters must be retrained. The auxiliary function where only terms dependent on the model parameters are shown is

$$\mathcal{Q}_J\left(\mathcal{M},\hat{\mathcal{T}};\hat{\mathcal{M}},\hat{\mathcal{T}}\right) = \\ -\frac{1}{2}\sum_{h=1}^{H}\sum_{t=1}^{T^{(h)}}\sum_{m=1}^{M}\gamma_t^{(m)}\sum_{i=1}^{D}\left(\log(\sigma_{s,i}^{(m)2}+\hat{\sigma}_{\mathtt{b},i}^{(r_m h)2})+\frac{(\hat{\boldsymbol{a}}_i^{(r_m h)}\boldsymbol{o}_t+\hat{b}_i^{(r_m h)}-\mu_{s,i}^{(m)})^2}{\sigma_{s,i}^{(m)2}+\hat{\sigma}_{\mathtt{b},i}^{(r_m h)2}}\right) \quad (101)$$

where diagonal covariance matrices assumed. Because the joint transform parameters affect the model parameters and are shared over many homogeneous blocks, there is no closed form solution for the model parameters that maximise this auxiliary function. Hence a generalised EM approach is taken, where Newton's method is applied to optimise the model parameters in the maximisation step. This requires both first and second derivatives of the auxiliary function with respect to the model mean and variance.

The first derivative of the auxiliary in equation 101 with respect to the mean of component m, dimension i is

$$\frac{\partial\mathcal{Q}_J}{\partial\mu_{s,i}^{(m)}} = \sum_{h=1}^{H}\sum_{t=1}^{T^{(h)}}\gamma_t^{(m)}\left(\frac{\hat{\boldsymbol{a}}_i^{(r_m h)}\boldsymbol{o}_t+\hat{b}_i^{(r_m h)}-\mu_{s,i}^{(m)}}{\sigma_{s,i}^{(m)2}+\hat{\sigma}_{\mathtt{b},i}^{(r_m h)2}}\right) \quad (102)$$

and with respect to the model variance

$$\frac{\partial\mathcal{Q}_J}{\partial\sigma_{s,i}^{(m)2}} = \sum_{h=1}^{H}\sum_{t=1}^{T^{(h)}}\frac{\gamma_t^{(m)}}{2\left(\sigma_{s,i}^{(m)2}+\hat{\sigma}_{\mathtt{b},i}^{(r_m h)2}\right)}\left(\frac{(\hat{\boldsymbol{a}}_i^{(r_m h)}\boldsymbol{o}_t+\hat{b}_i^{(r_m h)}-\mu_{s,i}^{(m)})^2}{\sigma_{s,i}^{(m)2}+\hat{\sigma}_{\mathtt{b},i}^{(r_m h)2}}-1\right) \quad (103)$$

From these derivatives it can be seen that when the noise is high, the uncertainty bias term $\hat{\sigma}_{\mathtt{b},i}^{(r_m h)2}$, which increases in noise, will de-weight the contribution of that observation. In areas where the noise completely subsumes the speech, the uncertainty will ensure that these observations do not contribute to the estimate of the model parameters at all. This allows the model parameters to be a better representation of "clean" speech.

The Hessian matrix is also required and is comprised of the following terms

$$\frac{\partial^2 \mathcal{Q}_J}{\partial \mu_{s,i}^{(m)2}} = \sum_{h=1}^{H} \frac{-1}{\sigma_{s,i}^{(m)2} + \hat{\sigma}_{\mathsf{b},i}^{(r_m h)2}} \sum_{t=1}^{T^{(h)}} \gamma_t^{(m)} \tag{104}$$

$$\frac{\partial^2 \mathcal{Q}_J}{\partial \left(\sigma_{s,i}^{(m)2}\right)^2} = \sum_{h=1}^{H} \frac{1}{\sigma_{s,i}^{(m)2} + \hat{\sigma}_{\mathsf{b},i}^{(r_m h)2}} \sum_{t=1}^{T^{(h)}} \gamma_t^{(m)} \left(\frac{1}{2} - \frac{\left(\hat{\boldsymbol{a}}_i^{(r_m h)} \boldsymbol{o}_t + \hat{b}_i^{(r_m h)} - \mu_{s,i}^{(m)}\right)^2}{\sigma_{s,i}^{(m)2} + \hat{\sigma}_{\mathsf{b},i}^{(r_m h)2}} \right) \tag{105}$$

$$\frac{\partial^2 \mathcal{Q}_J}{\partial \mu_{s,i}^{(m)} \partial \sigma_{s,i}^{(m)2}} = \frac{\partial^2 \mathcal{Q}_J}{\partial \sigma_{s,i}^{(m)2} \partial \mu_{s,i}^{(m)}} = \sum_{h=1}^{H} \frac{-1}{\sigma_{s,i}^{(m)2} + \hat{\sigma}_{\mathsf{b},i}^{(r_m h)2}} \sum_{t=1}^{T^{(h)}} \gamma_t^{(m)} \left(\frac{\hat{\boldsymbol{a}}_i^{(r_m h)} \boldsymbol{o}_t + \hat{b}_i^{(r_m h)} - \mu_{s,i}^{(m)}}{\sigma_{s,i}^{(m)2} + \hat{\sigma}_{\mathsf{b},i}^{(r_m h)2}} \right) \tag{106}$$

This gives the following update formula

$$\begin{bmatrix} \hat{\mu}_{s,i}^{(m)} \\ \hat{\sigma}_{s,i}^{(m)2} \end{bmatrix} = \begin{bmatrix} \mu_{s,i}^{(m)} \\ \sigma_{s,i}^{(m)2} \end{bmatrix} - \zeta \begin{bmatrix} \frac{\partial^2 \mathcal{Q}_J}{\partial \mu_{s,i}^{(m)2}} & \frac{\partial^2 \mathcal{Q}_J}{\partial \mu_{s,i}^{(m)} \partial \sigma_{s,i}^{(m)2}} \\ \frac{\partial^2 \mathcal{Q}_J}{\partial \sigma_{s,i}^{(m)2} \partial \mu_{s,i}^{(m)}} & \frac{\partial^2 \mathcal{Q}_J}{\partial \left(\sigma_{s,i}^{(m)2}\right)^2} \end{bmatrix}^{-1} \begin{bmatrix} \frac{\partial \mathcal{Q}_J}{\partial \mu_{s,i}^{(m)}} \\ \frac{\partial \mathcal{Q}_J}{\partial \sigma_{s,i}^{(m)2}} \end{bmatrix} \tag{107}$$

In practice, the stabilising learning rate ζ may be less than one, but in this work a value of unity is typically used. It was found that during this optimisation process, the variance may sometimes be driven to infinity. Hence another measure was introduced to stabilise the variance estimation—the variance was limited to only increase or decrease by a factor υ. Specifically

$$\hat{\sigma}_{s,i}^{(m)2} = \min\left(\max\left(\hat{\sigma}_{s,i}^{(m)2}, \frac{1}{\upsilon} \sigma_{s,i}^{(m)2} \right), \upsilon \sigma_{s,i}^{(m)2} \right) \tag{108}$$

In practice, υ was set at 2. To compute the terms of the Hessian matrix, the following statistics may be gathered per recognition component

$$w_{1,i}^{(m)} = \frac{\partial^2 \mathcal{Q}_J}{\partial \mu_{s,i}^{(m)2}} \tag{109}$$

$$w_{2,i}^{(m)} = \sum_{h=1}^{H} \sum_{t=1}^{T^{(h)}} \frac{\gamma_t^{(m)}}{\left(\sigma_{s,i}^{(m)2} + \sigma_{\mathsf{b},i}^{(r_m h)2}\right)^2} \tag{110}$$

$$w_{3,i}^{(m)} = \frac{\partial^2 \mathcal{Q}_J}{\partial \sigma_{s,i}^{(m)2} \partial \mu_i^{(m)}} \tag{111}$$

$$w_{4,i}^{(m)} = -\sum_{h=1}^{H} \sum_{t=1}^{T^{(h)}} \frac{\gamma_t^{(m)} (\boldsymbol{a}_i^{(r_m h)} \boldsymbol{o}_t + b_i^{(r_m h)} - \mu_{s,i}^{(m)})^2}{\left(\sigma_{s,i}^{(m)2} + \sigma_{\mathsf{b},i}^{(r_m h)2}\right)^3} \tag{112}$$

while the first order partial derivatives may be directly accumulated. The second order derivative with respect to the model variance is then given by

$$\frac{\partial^2 \mathcal{Q}_J}{\partial (\sigma_{s,i}^{(m)2})^2} = w_{4,i}^{(m)} + \frac{1}{2} w_{2,i}^{(m)} \tag{113}$$

When using this form of optimisation for maximising the auxiliary function, it is important to ensure that the iterations are approaching a global maximum rather than the minimum. This implies that the Hessian matrix must be negative definite and necessitates checking the second derivatives, given by equations 104 and 105, are negative; the former is guaranteed to always

be, however the latter is not so well conditioned. This may be simply done by re-expressing equation 113 as

$$\frac{\partial^2 \mathcal{Q}_J}{\partial (\sigma_{s,i}^{(m)2})^2} = w_{2,i}^{(m)} \left(\frac{w_{4,i}^{(m)}}{w_{2,i}^{(m)}} + \frac{1}{2} \right) \tag{114}$$

$$= w_{2,i}^{(m)} \left(-\hat{\vartheta} + \frac{1}{2} \right) \tag{115}$$

where

$$\hat{\vartheta} = \max \left(\vartheta, -\frac{w_{4,i}^{(m)}}{w_{2,i}^{(m)}} \right) \tag{116}$$

This parameter ϑ should remain greater than a half to ensure stability of the optimisation. It may be observed that the ratio of $w_{4,i}^{(m)}$ to $w_{2,i}^{(m)}$ should converge to unity as the model parameters better approximate the training data, given the set of `Joint` transforms.

Instead of directly optimising the variance, the log of the variance may be estimated to ensure that the converged value remains positive. Thus, make the change of variable

$$\boldsymbol{\varsigma}_s^{(m)} = \log \boldsymbol{\Sigma}_s^{(m)} \tag{117}$$

which changes the parameter update formula, stated in equation 107, to

$$\begin{bmatrix} \hat{\mu}_{s,i}^{(m)} \\ \hat{\varsigma}_{s,i}^{(m)} \end{bmatrix} = \begin{bmatrix} \mu_{s,i}^{(m)} \\ \varsigma_{s,i}^{(m)} \end{bmatrix} - \zeta \begin{bmatrix} \frac{\partial^2 \mathcal{Q}_J}{\partial \mu_{s,i}^{(m)2}} & \frac{\partial^2 \mathcal{Q}_J}{\partial \mu_{s,i}^{(m)} \partial \varsigma_{s,i}^{(m)}} \\ \frac{\partial^2 \mathcal{Q}_J}{\partial \varsigma_{s,i}^{(m)} \partial \mu_{s,i}^{(m)}} & \frac{\partial^2 \mathcal{Q}_J}{\partial \varsigma_{s,i}^{(m)2}} \end{bmatrix}^{-1} \begin{bmatrix} \frac{\partial \mathcal{Q}_J}{\partial \mu_{s,i}^{(m)}} \\ \frac{\partial \mathcal{Q}_J}{\partial \varsigma_{s,i}^{(m)}} \end{bmatrix} \tag{118}$$

The partial derivatives with respect to $\boldsymbol{\varsigma}_{s,i}^{(m)}$ may be expressed as a function of the previously given partial derivatives

$$\frac{\partial \mathcal{Q}_J}{\partial \varsigma_{s,i}^{(m)}} = \frac{\partial \sigma_{s,i}^{(m)2}}{\partial \varsigma_{s,i}^{(m)}} \frac{\partial \mathcal{Q}_J}{\partial \sigma_{s,i}^{(m)2}} \tag{119}$$

$$= \sigma_{s,i}^{(m)2} \frac{\partial \mathcal{Q}_J}{\partial \sigma_{s,i}^{(m)2}} \tag{120}$$

$$\frac{\partial^2 \mathcal{Q}_J}{\partial \mu_{s,i}^{(m)} \partial \varsigma_{s,i}^{(m)}} = \frac{\partial^2 \mathcal{Q}_J}{\partial \varsigma_{s,i}^{(m)} \partial \mu_{s,i}^{(m)}} = \frac{\partial \sigma_{s,i}^{(m)2}}{\partial \varsigma_{s,i}^{(m)}} \frac{\partial^2 \mathcal{Q}_J}{\partial \sigma_{s,i}^{(m)2} \partial \mu_{s,i}^{(m)}} \tag{121}$$

$$= \sigma_{s,i}^{(m)2} \frac{\partial^2 \mathcal{Q}_J}{\partial \sigma_{s,i}^{(m)2} \partial \mu_{s,i}^{(m)}} \tag{122}$$

since $\frac{\partial \sigma_{s,i}^{(m)2}}{\partial \varsigma_{s,i}^{(m)}} = \exp \varsigma_{s,i}^{(m)} = \sigma_{s,i}^{(m)2}$, and lastly

$$\frac{\partial^2 \mathcal{Q}_J}{\partial \varsigma_{s,i}^{(m)2}} = \frac{\partial}{\partial \varsigma_{s,i}^{(m)}} \left\{ \frac{\partial \sigma_{s,i}^{(m)2}}{\partial \varsigma_{s,i}^{(m)}} \frac{\partial \mathcal{Q}_J}{\partial \sigma_{s,i}^{(m)2}} \right\} \tag{123}$$

$$= \left\{ \frac{\partial}{\partial \varsigma_{s,i}^{(m)}} \frac{\partial \sigma_{s,i}^{(m)2}}{\partial \varsigma_{s,i}^{(m)}} \right\} \frac{\partial \mathcal{Q}_J}{\partial \sigma_{s,i}^{(m)2}} + \frac{\partial \sigma_{s,i}^{(m)2}}{\partial \varsigma_{s,i}^{(m)}} \left\{ \frac{\partial}{\partial \varsigma_{s,i}^{(m)}} \frac{\partial \mathcal{Q}_J}{\partial \sigma_{s,i}^{(m)2}} \right\} \tag{124}$$

$$= \left\{ \frac{\partial \sigma_{s,i}^{(m)2}}{\partial \varsigma_{s,i}^{(m)}} \right\} \frac{\partial \mathcal{Q}_J}{\partial \sigma_{s,i}^{(m)2}} + \frac{\partial \sigma_{s,i}^{(m)2}}{\partial \varsigma_{s,i}^{(m)}} \left\{ \frac{\partial \sigma_{s,i}^{(m)2}}{\partial \varsigma_{s,i}^{(m)}} \frac{\partial^2 \mathcal{Q}_J}{\partial (\sigma_{s,i}^{(m)2})^2} \right\} \tag{125}$$

$$= \sigma_{s,i}^{(m)2} \frac{\partial \mathcal{Q}_J}{\partial \sigma_{s,i}^{(m)2}} + \left(\sigma_{s,i}^{(m)2} \right)^2 \frac{\partial^2 \mathcal{Q}_J}{\partial (\sigma_{s,i}^{(m)2})^2} \tag{126}$$

6 Experiments

The compensation techniques discussed here are evaluated on two corpora: an artificially corrupted 1000 word Resource Management (RM) task, and the large vocabulary Broadcast News (BN) database of found audio. The effectiveness of VTS compensation, model-based `Joint` uncertainty decoding, their respective noise estimation methods, `Joint` adaptive training, and comparing clean and multistyle acoustic model training was first investigated on RM. The smaller vocabulary size and artificial nature makes RM an ideal corpus for conducting controlled experiments at a variety of noise levels and on clean and multistyle acoustic models. RM is however an artificial task, and it is important to apply these techniques to real found data. Hence, experiments were also conducted on the Broadcast News corpus.

All the systems evaluated used MFCC parameters with the 0^{th} cepstra, computed from the log magnitude spectrum, and associated first- and second-order dynamic features computed using linear regression producing a 39-dimensional feature vector. Experiments were conducted using an internal version of HTK 3.3 [44] with additional routines to support `Joint` and VTS noise parameter estimation, compensation within the adaptation framework, and `Joint` adaptive training. Systems referred to as "matched" are estimated using single-pass re-training (SPR) [12, 44]. This gives a suitable upper limit for the performance of model-based noise compensation schemes where it is assumed only the output distributions need to be updated to reflect the environmental noise, and the state transitions or component weights need not be re-estimated. Performance figures reported are % word error rates (WER).

6.1 Resource Management

The noise robustness techniques discussed were first evaluated on a simple task to gauge their effectiveness before attempting a more challenging task. The Resource Management database is a balanced 1000 word vocabulary command and control task [34]. To create different noise conditions, the RM task was corrupted with Operations Room noise from the NOISEX-92 database at the waveform level; this noise has a dominant low frequency background hum, and some intermittent speech and machine noise. For testing, results are averaged across the `Feb'89`, `Oct'89` and `Feb'91` test sets totaling an hour of speech, 30 speakers and 900 test utterances. Results are mostly reported on the 20 and 14 dB SNR conditions as they provided a balance between difficulty and reasonable performance. For all the RM experiments using `Joint` uncertainty decoding, 16 diagonal transforms was the default configuration. No cepstral mean normalisation was performed.

The acoustic models are state-clustered, cross-word triphone models yielding 1582 states, each with 6 components, giving a total of 9492 system Gaussians. The language model provides simple bigram word-pair probabilities. This corresponds closely to the RM recipe provided with HTK. Three forms of acoustic models were evaluated. Clean models were trained on the original RM data which was recorded at 47 dB SNR, encompassing 3990 sentences from 109 speakers. Multistyle trained models trained on the same amount of data however at five levels of SNR from 32 dB to 8 dB, randomly changing from speaker to speaker; no actual clean data was included since such high SNR audio rarely exists in practice. A JAT acoustic model was trained using four full interleaved steps starting from the multistyle acoustic model: one step involved first training the `Joint` transforms, and then four iterations of acoustic model parameter updates with the variance Hessian floor parameter ϑ diminishing from 2.5 to 1.0 in 0.5 increments. When test transforms are needed, these are estimated for each iteration of models, and used as input transforms for the next, until transforms are estimated for the test conditions for the final canonical model.

6.1.1 Compensation Baselines

First, some baseline performance of the predictive noise compensation schemes discussed in this report are presented. Table 1 summarises the performance of these schemes, using both clean and multistyle trained acoustic models, and different forms of noise estimation. As expected, an uncompensated ASR system clearly performs poorly when noise is present as demonstrated by

Acoustic Model	Compensation	Noise Est. Type	Test Set SNR		
			Clean	20 dB	14 dB
Clean	—	—	3.1	38.0	83.7
	`Joint`	VTS	3.1	10.1	35.3
		`Joint`	3.1	9.2	22.6
	VTS	VTS	3.0	8.4	23.6
Multistyle	—	—	11.7	7.0	15.5
	`Joint`	VTS	9.0	8.6	15.9
		`Joint`	8.6	6.7	12.3
	VTS	VTS	8.8	6.5	12.0
Matched	—	—	3.1	7.4	14.3

Table 1: RM Compensation, clean and multistyle models, varying noise estimation scheme.

large error rates of 38.0% at 20 dB and 83.7% at 14 dB. Noise models were estimated using only one complete iteration over the complete data set with the initial supervision hypothesis. Despite the poor performance of the clean acoustic models, using these results to supervise the noise model estimation still gave good results. At 20 dB, the 38.0% was reduced to a 9.2% error rate for `Joint` and 8.4% for VTS; this compares well to the 7.4% for the matched system. In the clean test condition, these compensation forms simply provide an ML estimate of the channel noise which may also capture some speaker differences; this provides a negligible improvement on the clean system with the WER unchanging at 3.1%.

Multistyle trained models, even without compensation, give better results than compensated clean acoustic models, and perform comparably to the matched systems: 7.0% compared to 7.4% at 20 dB and 15.5% to 14.3% at 14 dB. On clean test data, the multistyle acoustic models exhibit significant degradation, and even with noise compensation, is still far from clean on clean performance of 3.1%. This is due to the multistyle training data not including actual "clean" speech, and thus this condition is completely unseen. Applying VTS or `Joint` compensation on the noisier tests gave even further improvements exceeding matched performance. These results show that such predictive noise compensation schemes can be effectively applied to multistyle trained acoustic models.

Two other points are demonstrated in table 1. The first, is that model-based `Joint` compensation performance is similar to VTS compensation with both clean and multistyle acoustic models. At 14 dB SNR, the WER% of 22.6 is likely a noisy figure as it should not be better than the VTS at 23.6. Comparing the multistyle systems, the `Joint` and VTS systems are very similar: 6.7 compared to 6.5% at 20 dB and 12.3 to 12.0% at 14 dB. The second aspect shown is that, for `Joint` uncertainty decoding, `Joint` noise estimation gives better results than using the noise estimates provided by VTS-based noise estimation. This result is exaggerated on the multistyle systems, for example at 14 dB, the `Joint` performance degrades using the VTS noise estimates (15.9%) over no compensation (15.5%), but significantly improves the result when `Joint` noise estimates are used (12.3%). Thus it is clear, for all conditions tested, that the type of noise estimation should match the compensation form for best results.

6.1.2 Estimation of Noise Parameters

There is a significant difference in accuracy of the supervision hypothesis between the clean and multistyle systems: an error rate of 38.0% at 20 dB for example compared to 7.0% for the equivalent multistyle. Hence the comparison between noise compensation techniques using different model training methods may be considered unfair. Since the full back-end acoustic model is used in the noise estimation process, using compensated models to compute the state alignments may give improvements. The accuracy of the supervision hypothesis may also be a factor in the quality of the noise estimates. Hence two approaches are examined to refine the noise estimates for the

clean systems. The first conducts a second EM iteration over the complete dataset, but without updating the hypothesis; this gives an indication of the importance of having models that better match the noisy condition when estimating noise parameters. The second updates the hypothesis and performs an EM step; this is more expensive than the first as decoding must also be performed to obtain the updated supervision in addition to re-aligning the models to the data. Comparing the outcome of these two approaches can give insight of the sensitivity of the noise estimation process to the accuracy of the hypothesis.

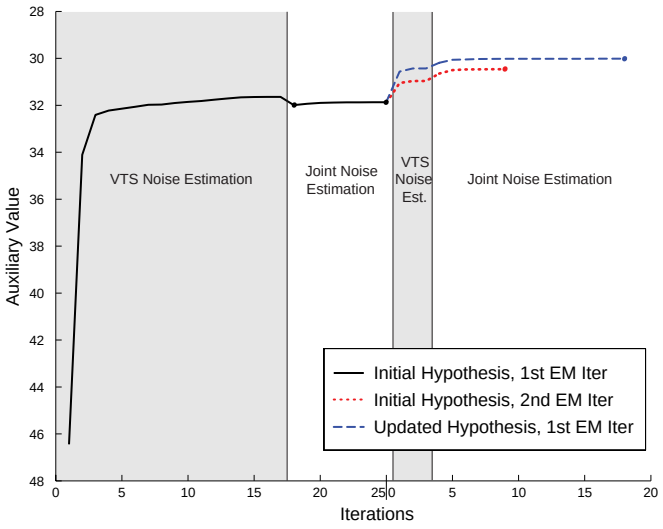

Figure 7: Auxiliary function value during noise estimation.

Figure 7 plots the auxiliary function value for the estimation of the noise at 20 dB for a speaker. There are three series: the first is the first EM iteration to estimate the noise, where the models are uncompensated, the hypothesis from this uncompensated system is used to produce the alignments, and the initial noise parameters are set to the minimum energy vector for the additive noise mean, the variance to the first five frames of speech, and channel noise to zero. There are large gains in the auxiliary with the first few iterations updating the VTS expansion point; this is step 5 in figure 5, which does not go over the complete dataset again. The VTS noise estimation is considered to have converged after 17 iterations, and another 8 iterations of `Joint` noise estimation is conducted. There is a drop in the auxiliary function as we shift from using a ML VTS noise estimation and VTS compensation for the auxiliary to using this noise estimate and `Joint` compensation. Then sufficient statistics are gathered again over the entire dataset using this new noise estimate to produce `Joint` transforms to compensate the models used to align the data. This can be done without updating the hypothesis (step 6 from figure 5, shown in red, and with an update, shown in blue (step 7). Here the preceding VTS noise estimation changes the auxiliary value very little, and now that the noise estimation is tuned to the `Joint` auxiliary function there is an improvement of the auxiliary function over the VTS auxiliary value. Subsequent `Joint` noise estimation iterations increase the auxiliary slightly.

Table 2 demonstrates how only slight gains are attained when the hypothesis is updated and that the second EM iteration provides most of the improvement over a single EM iteration. This is the case for both VTS and `Joint` compensation on clean acoustic models. The results were similar at an SNR of 14 dB. Note that at this level of noise the first hypothesis has an error rate of 38%, yet relatively accurate noise estimates were still attained since the performance of VTS or `Joint` compensation using these estimates is still good; this indicates a lack of sensitivity to the hypothesis. But by updating the hypothesis, at this noise level, the VTS performance reaches the matched. This is not the case at a noise level of 14 dB; the VTS performance, with a second iteration and updated hypothesis gives an error rate of 21.2% which is far from the matched level

Acoustic Model	Compensation	Hypo	Iteration 1	Iteration 2
Clean	—		38.0	
	`Joint`	1	9.2	8.6
		2	8.4	—
	VTS	1	8.4	7.7
		2	7.4	—
Matched	—		7.4	

Table 2: RM compensation, clean trained acoustic models tested at 20 dB SNR.

of 14.3%.

6.1.3 Amount of Estimation Data

In the previous experiments, a noise model was estimated for each speaker, which totals approximately ten thousand frames per speaker. It is interesting to investigate how the noise estimation process performs when less data is available and compare this to the estimation of CMLLR transforms. There are significantly more free parameters to train 16 diagonal CMLLR transforms, each transform having 2×39 parameters, compared to just estimating the noise parameters, which has $13+13+39$ parameters, as shown in table 3. When the amount of data for estimation is reduced

System	Number of Parameters	Amount of data 1 Utt	Amount of data 30 Utts
CMLLR	1248	14.7	13.1
`Joint`	65	13.3	12.3
VTS		13.3	12.0

Table 3: RM compensation, multistyle acoustic models varying amount of adaptation data at 14 dB SNR.

to one utterance, about 300 frames, all systems degrade similarly. The relative reduction in error is 11% for the 16 diagonal transform CMLLR, when the amount of data is increased, and 8% and 10% for the 16 transform `Joint` system and VTS compensation respectively. While 16 diagonal CMLLR transforms may still be estimated robustly with just an utterance, it is can be seen that regardless of the amount of data, the addition of the variance bias term gives a definite gain in accuracy.

6.1.4 `Joint` Adaptive Training

A `Joint` adaptive training framework was presented in section 5 as an alternative form of training to yield the "clean" acoustic models necessary for the noise compensation schemes discussed. Table 4 compares JAT with multistyle and matched training. It is clear that JAT is superior to multistyle systems compensated with `Joint` transforms; for the clean environment, there is a gain from 8.6% error to 5.7%, at 20 dB from 6.7% to 6.2% and at 14 dB a 0.8% improvement. This table also shows how `Joint` transforms can complement CMLLR transforms; here two full matrix CMLLR transforms are used to compensate the multistyle system, or in conjunction with `Joint` parent transforms. Adding CMLLR reduces the remaining mismatch between testing and training conditions and models correlations between different dimensions. This complements the `Joint` transforms which account for the environmental noise yielding the best systems for the noisy conditions at 5.7% WER at 20 dB and 10.9% at 14 dB. This exceeds the matched performance. Despite having no clean data in the training data, clean performance is also significantly improved

Acoustic Model	Compensation		Test Set SNR		
	`Joint`	CMLLR	Clean	20 dB	14 dB
Multistyle			11.7	7.0	15.5
		✓	5.0	5.7	13.8
	✓		8.6	6.7	12.3
	✓	✓	5.4	5.8	11.7
JAT	✓		5.7	6.2	11.4
	✓	✓	4.7	5.7	10.9
Matched			3.1	7.4	14.3

Table 4: RM compensation, multistyle trained versus JAT acoustic models with 256 diagonal `Joint` and 2 full CMLLR transforms.

to 4.7%, compared to a best of 5.4% for the multistyle system, however is still far from the 3.1% for matched. These results show that `Joint` adaptive training is superior to multistyle training.

Interestingly enough, the CMLLR compensation is able to bring down the multistyle WER significantly, for example from 11.7% to 5.0% on clean test data; this indicates the multistyle training yields models that deviate significantly from representing clean speech, but may be simply compensated by two full linear transforms. However more significant is the performance of the JAT acoustic models. The JAT acoustic model showed improvements over using just a multistyle acoustic model from 6.7% to 6.2% at 20dB and 12.3% to 11.4% at 14dB. Adding two full CMLLR transforms, using the `Joint` transforms as parent transforms on the JAT acoustic model, gave the best overall system performance, on noisy data, at 5.7% and 10.9%, exceeding the performance of both matched and the most powerfully compensated multistyle systems.

6.2 Broadcast News

These positive results on RM motivate testing on more challenging task such as Broadcast News. The BN system used here is based on a simplified version of the CU-HTK RT-03 BN-E system [19]. The acoustic model is trained in an ML fashion, as opposed to MPE, on approximately 143 hours of found data from recorded English Broadcast News released by the LDC in 1997 and 1998. State-tied cross-word tri-phone models were defined using decision tree clustering. This gave approximately 7000 states, with each state modeled by 16 Gaussians, yielding about 110k acoustic model components. Testing was conducted on the `dev-03` test set totaling 3 hours of shows from six different news sources aired in January, 2001. The same segmentation and clustering routines from the RT-03 system were used to provide homogeneous blocks of training and test data. MFCC parameters were chosen over PLP, without CMN. Decoding was done without adaptation, using a 59k-word dictionary and a bigram language model to generate lattices. A trigram language model is then used to re-score the lattices and find the 1-best transcription. Only recognition on wide-band data was compensated using the techniques discussed; the same narrow-band results, from an uncompensated system, were used during scoring for all the systems described. A more complex system would typically use some form feature projection scheme such as HLDA [22] or fMPE [32], advanced covariance modeling such as STC [11], and MMI [42] or MPE [33] training of model parameters; the use of such techniques were not investigated in these experiments.

Results for a noise compensated Broadcast News system are presented in table 5. The test set itself is not very noisy, with large amounts of studio audio. The baseline WER for the uncompensated multistyle BN system is 20.8%. Conducting adaptive training using CMN reduces this to 19.4%. Using only 16 `Joint` transforms, estimated using `Joint` noise estimation as in the RM system, does not give gains over this, however increasing the number to 256, to reflect the magnitude increase in acoustic model components, does. With 256 `Joint` transforms, the WER of 18.8% is equivalent to the more computationally demanding VTS-style compensation; using ML VTS noise estimates degrades this to 19.1% emphasising the need to tailor the noise estimation

Test Set	Compensation	Number of Transforms	%WER
dev-03	—	—	20.8
	Joint	16	19.4
		256	18.8
	VTS	—	18.8

Table 5: Noise compensation on Broadcast News.

form to the compensation. As seen on the RM task, these predictive noise compensation schemes complement linear transforms. Adapting with two full CMLLR transforms yields an error rate of 18.1%, but combined with VTS compensation or 256 Jointtransforms gives an overall WER of 17.7%. This reinforces findings on the RM database that suggest applying predictive noise compensation schemes, such as VTS or Joint uncertainty decoding, to large multistyle trained acoustic models is beneficial. Initial experiments with JAT on BN data did not show any gains; this was felt to be due to the low amount of noise present in the training data.

7 Conclusions

This report has investigated the use of model-based VTS compensation and `Joint` uncertainty decoding with clean and multistyle trained acoustic models, in unseen noise conditions, and with artificially corrupted and real noisy data. A straightforward noise estimation scheme is presented to estimate ML cepstral models of the additive and convolutional noise based on an EM framework for either VTS or `Joint` compensation. Not only are the static features of the additive noise variance estimated, the dynamic ones are as well; this allows the dynamic features and variances of the acoustic model to be compensated. Furthermore, this scheme permits the noise model to be continuously updated, even during long segments of speech, while simpler methods of estimating noise from background regions cannot. On an artificially corrupted, medium vocabulary RM task, this noise estimation scheme gave results close to matched performance with either VTS or `Joint` transforms compensating clean acoustic models. However, this scheme may also be thought of as estimating a ML noise model that produces transforms which best reduce the mismatch between training and test conditions; hence it may also be applied with multistyle trained models. Experiments showed that using these predictive techniques on such a model was more effective than on a clean trained one, and exceeded matched system performance.

It was also demonstrated that model-based `Joint` uncertainty decoding may be viewed as an approximation to VTS compensation. Instead of computing the VTS approximation for every acoustic model component, the `Joint` form allows the approximation to be shared over a class or cluster of similar components. This improves efficiency with only a small loss in accuracy and allows the `Joint` form to scale performance to the available compuatation resources. Also, these predictive techniques, while better than linear transforms in compensating for noise, were also complementary to CMLLR and can be effectively combined. A linear transform of features can only address low levels of noise, whereas the predictive techniques discussed also update model variances such that they can cope with the mismatch of higher levels of noise. However, since the VTS and `Joint` compensation forms used only noise mismatch functions, the remaining mismatch due to other factors can be addressed by CMLLR. These results were clearly shown on both clean and multistyle RM systems and on the large vocabulary BN multistyle system.

The report also introduced a `Joint` adaptive training framework as a more effective approach to handling noise in the training data compared to multistyle training. This form of adaptive training explicitly normalises out noise to yield a "clean" acoustic model whereas multistyle training includes the variation due to noise in the models. A generalised EM method and update formula for estimating the `Joint` transforms and canonical model mean and variance parameters was provided. There were no closed form solutions for the latter, hence a generic iterative optimisation was given along with several mechanisms to stabilise the process. A feature of JAT is that the uncertainty bias added to the model variances effectively de-weights observations that are noisy. This allows JAT to handle a wide range of SNR in the training data, while maintaining a noise-free acoustic model. Results for JAT were reported on the RM task. Adapted from a multistyle acoustic model, the JAT system consistently gave a small gain in performance over applying compensation techniques directly to the multistyle system. While the multistyle systems performed quite poorly on clean data, which was not present in the model training data, the JAT acoustic model was far better at handling this unseen condition. Hence, it may be concluded that JAT is superior to multistyle training on noisy training data.

Acknowledgements

Hank Liao would like to thank Toshiba Research Europe Ltd. for funding this work.

References

[1] A. Acero, L. Deng, T. T. Kristjansson, and J. Zhang. HMM adaptation using vector Taylor series for noisy speech recognition. In *Proc. ICSLP*, Beijing, China, Oct. 2000.

[2] T. Anastasakos, J. McDonough, R. Schwartz, and J. Makhoul. A compact model for speaker-adaptive training. In *Proc. ICSLP*, 1996.

[3] C. Benítez, J. C. Segura, A. de la Torre, , J. Ramírez, and A. J. Rubio. Including uncertainty of speech observations in robust speech recognition. In *Proc. ICSLP*, Jeju Island, Korea, Oct. 2004.

[4] S. F. Boll. Suppression of acoustic noise in speech using spectral subtraction. *IEEE Transactions ASSP*, 27:113–120, 1979.

[5] L. Deng, A. Acero, M. Plumpe, and X. D. Huang. Large vocabulary speech recognition under adverse acoustic environments. In *Proc. ICSLP*, pages 806–809, Beijing, China, Oct. 2000.

[6] L. Deng, J. Droppo, and A. Acero. Recursive estimation of nonstationary noise using iterative stochastic approximation for robust speech recognition. *IEEE Trans. on Speech and Audio Processing*, 11(6), Nov. 2003.

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

[8] J. Droppo, L. Deng, and A. Acero. Evaluation of the SPLICE algorithm on the Aurora 2 database. In *Proc. of Eurospeech 2001*, pages 217–220, Aalborg, Denmark, Sept. 2001.

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

[10] B. Frey, T. T. Kristjansson, L. Deng, and A. Acero. ALGONQUIN – learning dynamic noise models from noisy speech for robust speech recognition. In *Proc NIPS*, 2001.

[11] M. Gales. Semi-tied covariance matrices for hidden Markov models. *IEEE Transactions on Speech and Audio Processing*, May 1999.

[12] M. J. F. Gales. *Model-Based Techniques for Noise Robust Speech Recognition*. PhD thesis, Cambridge University, 1995.

[13] M. J. F. Gales. The generation and the use of regression class trees for MLLR adaptation. Technical Report CUED/F-INFENG/TR263, University of Cambridge, 1996. Available via anonymous ftp from: svr-www.eng.cam.ac.uk.

[14] M. J. F. Gales. Maximum Likelihood Linear Transformations For HMM-Based Speech Recognition. *Computer Speech and Language*, 12, Jan. 1998.

[15] M. J. F. Gales. Predicative model based compensation schemes for robust speech recognition. *Speech Communication*, 25, 1998.

[16] M. J. F. Gales. Acoustic factorisation. In *Proc. ASRU*, 2001.

[17] H. Hermansky and N. Morgan. RASTA processing of speech. *IEEE Transactions on Speech and Audio Processing*, 2(4), Oct. 1994.

[18] H.-G. Hirsch and D. Pearce. The AURORA experimental framework for the evaluation of speech recognition systems under noisy conditions. In *Proc. ASR-2000*, pages 181–188, Sept. 2000.

[19] D. Y. Kim, G. Evermann, T. Hain, D. Mrva, S. E. Tranter, L. Wang, and P. C. Woodland. Recent advances in broadcast news transcription. In *Proc. ASRU*, 2003.

[20] D. Y. Kim, N. S. Kim, and C. K. Un. Model-based approach for robust speech recognition in noisy environments with multiple noise sources. In *Proc. Eurospeech*, Rhodes, Greece, Sept. 1997.

[21] D. Y. Kim, C. K. Un, and N. S. Kim. Speech recognition in noisy environments using first-order vector Taylor series. *Speech Communication*, 24(1):39–49, June 1998.

[22] N. Kumar. *Investigation of Silicon Auditory Models and Generalization of Linear Discriminant Analysis for Improved Speech Recognition*. PhD thesis, John Hopkins University, Baltimore, Maryland, 1997.

[23] C. Leggetter and P. C. Woodland. Maximum likelihood linear regression for speaker adaptation of continuous density HMMs. *Computer Speech and Language*, 9:171–186, 1995.

[24] H. Liao and M. J. F. Gales. Uncertainty decoding for noise robust speech recognition. Technical Report CUED/F-INFENG/TR499, University of Cambridge, 2004. Available from: mi.eng.cam.ac.uk/~hl251.

[25] H. Liao and M. J. F. Gales. Joint uncertainty decoding for noise robust speech recognition. In *Proc. Interspeech*, 2005.

[26] H. Liao and M. J. F. Gales. Issues with uncertainty decoding for noise robust speech recognition. In *Proc. Interspeech*, 2006.

[27] H. Liao and M. J. F. Gales. Issues with uncertainty decoding for noise robust speech recognition. Technical Report CUED/F-INFENG/TR549, University of Cambridge, 2006. Available from: mi.eng.cam.ac.uk/~hl251.

[28] R. Lippman, E. A. Martin, and D. B. Paul. Multi-style training for robust isolated-word speech recognition. In *Proc. ICASSP*, 1987.

[29] H. Mehanna. Estimating noise models using noise corrupted data. Master's thesis, University of Cambridge, Cambridge, UK, July 2004.

[30] P. Moreno. *Speech Recognition in Noisy Environments*. PhD thesis, Carnegie Mellon University, 1996.

[31] L. R. Neumeyer, A. Sankar, and V. V. Digalakis. A comparative study of speaker adaptation techniques. In *Proc. Eurospeech*, 1995.

[32] D. Povey, B. Kingsbury, L. Mangu, G. Saon, H. Soltau, and G. Zweig. fMPE: Discriminatively trained features for speech recognition. In *Proc. ICASSP*, 2005.

[33] D. Povey and P. Woodland. Minimum phone error and i-smoothing for improved discriminative training. In *Proc. ICASSP*, 2002.

[34] P. Price, W. M. Fisher, J. Bernstein, and D. S. Pallett. The DARPA 1000-word resource management database for continuous speech recognition. In *Proc. ICASSP*, 1988.

[35] J. C. Segura, M. C. Benítez, , A. de la Torre, S. Dupont, and A. J. Rubio. VTS residual noise compensation. In *Proc. ICSLP*, 2002.

[36] K. Shinoda and T. Watanabe. Speaker adaptation with autonomous control using tree structure. In *Proc. Eurospeech*, 1995.

[37] R. Stern, A. Acero, F.-H. Liu, and Y. Ohshima. *Signal processing for robust speech recognition*, pages 357–384. Kluwer Academic, 1997.

[38] V. Stouten, H. V. hamme, K. Demuynck, and P. Wambacq. Robust speech recognition using model-based feature enhancement. In *Proc. European Conference on Speech Communication and Technology*, pages 17–20, Geneva, Switzerland, Sept. 2003.

[39] V. Stouten, H. V. hamme, J. Duchateau, and P. Wambacq. Evaluation of model-based feature enhancement on the AURORA-4 task. In *Proc. European Conference on Speech Communication and Technology*, pages 349–352, Geneva, Switzerland, Sept. 2003.

[40] V. Stouten, H. V. hamme, and P. Wambacq. Joint removal of additive and convolutional noise with model-based feature enhancement. In *Proc. ICASSP*, 2004.

[41] V. Stouten, H. V. hamme, and P. Wambacq. Model-based feature enhancement with uncertainty decoding for noise robust ASR. *Speech Communication*, 2006.

[42] V. Valtchev, J. J. Odell, P. C. Woodland, and S. J. Young. MMIE training of large vocabulary recognition systems. *Speech Communication*, 22:303–314, June 1997.

[43] A. Varga, R. Moore, J. Bridle, K. Ponting, and M. Russell. Noise compensation algorithms for use with hidden Markov model based speech recognition. In *Proc. ICASSP*, 1988.

[44] S. J. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. C. Woodland. *The HTK Book (for HTK Version 3.3)*. University of Cambridge, Mar. 2004.

A Compensating Dynamic Coefficients using VTS

In state of the art recognition systems, time derivatives improve performance by addressing the continuous nature of speech. The Continuous-Time approximation has been used to give an analytic approximation of the first- and second-order dynamic features. Applying the chain rule to the first time derivative of the corrupted speech yields the following

$$\frac{\partial y_i}{\partial t} = \boldsymbol{\nabla}_{\mathbf{x}}\, y_i \cdot \frac{\partial \boldsymbol{x}}{\partial t} + \boldsymbol{\nabla}_{\mathbf{n}}\, y_i \cdot \frac{\partial \boldsymbol{n}}{\partial t} + \boldsymbol{\nabla}_{\mathbf{h}}\, y_i \cdot \frac{\partial \boldsymbol{h}}{\partial t} \tag{127}$$

Substituting the first-order VTS approximation in equation 6 with respect to time for the actual corrupted speech, gives the following

$$\frac{\partial y_i}{\partial t} \approx \frac{\partial y_{vts,i}}{\partial t} \tag{128}$$

$$= \boldsymbol{\nabla}_{\mathbf{x}} \left\{ y_i\big|_{\mu_0} + \boldsymbol{\nabla}_{\mathbf{x}}\, y_i\big|_{\mu_0} \cdot \left(\boldsymbol{x} - \boldsymbol{\mu}_x^{(m)}\right) + \boldsymbol{\nabla}_{\mathbf{n}}\, y_i\big|_{\mu_0} \cdot (\boldsymbol{n} - \boldsymbol{\mu}_n) + \boldsymbol{\nabla}_{\mathbf{h}}\, y_i\big|_{\mu_0} \cdot (\boldsymbol{h} - \boldsymbol{\mu}_h) \right\} \cdot \frac{\partial \boldsymbol{x}}{\partial t} + \tag{129}$$

$$\boldsymbol{\nabla}_{\mathbf{n}} \left\{ y_i\big|_{\mu_0} + \boldsymbol{\nabla}_{\mathbf{x}}\, y_i\big|_{\mu_0} \cdot \left(\boldsymbol{x} - \boldsymbol{\mu}_x^{(m)}\right) + \boldsymbol{\nabla}_{\mathbf{n}}\, y_i\big|_{\mu_0} \cdot (\boldsymbol{n} - \boldsymbol{\mu}_n) + \boldsymbol{\nabla}_{\mathbf{h}}\, y_i\big|_{\mu_0} \cdot (\boldsymbol{h} - \boldsymbol{\mu}_h) \right\} \cdot \frac{\partial \boldsymbol{n}}{\partial t} +$$

$$\boldsymbol{\nabla}_{\mathbf{h}} \left\{ y_i\big|_{\mu_0} + \boldsymbol{\nabla}_{\mathbf{x}}\, y_i\big|_{\mu_0} \cdot \left(\boldsymbol{x} - \boldsymbol{\mu}_x^{(m)}\right) + \boldsymbol{\nabla}_{\mathbf{n}}\, y_i\big|_{\mu_0} \cdot (\boldsymbol{n} - \boldsymbol{\mu}_n) + \boldsymbol{\nabla}_{\mathbf{h}}\, y_i\big|_{\mu_0} \cdot (\boldsymbol{h} - \boldsymbol{\mu}_h) \right\} \cdot \frac{\partial \boldsymbol{h}}{\partial t}$$

$$= \boldsymbol{\nabla}_{\mathbf{x}} \left\{ \boldsymbol{\nabla}_{\mathbf{x}}\, y_i\big|_{\mu_0} \cdot \boldsymbol{x} \right\} \cdot \frac{\partial \boldsymbol{x}}{\partial t} + \boldsymbol{\nabla}_{\mathbf{n}} \left\{ \boldsymbol{\nabla}_{\mathbf{n}}\, y_i\big|_{\mu_0} \cdot \boldsymbol{n} \right\} \cdot \frac{\partial \boldsymbol{n}}{\partial t} + \boldsymbol{\nabla}_{\mathbf{h}} \left\{ \boldsymbol{\nabla}_{\mathbf{h}}\, y_i\big|_{\mu_0} \cdot \boldsymbol{h} \right\} \cdot \frac{\partial \boldsymbol{h}}{\partial t} \tag{130}$$

Recall that the vector gradient quantity in the dot products, such as $\boldsymbol{\nabla}_{\mathbf{x}}\, y_i\big|_{\mu_0}$, is the gradient of the corrupted speech, with respect to $\boldsymbol{x}$, but with variables evaluated at $\boldsymbol{\mu}_0$ and hence is no longer a function of any of the random variables. Thus

$$\frac{\partial y_i}{\partial t} \approx \boldsymbol{\nabla}_{\mathbf{x}}\, y_i\big|_{\mu_0} \cdot \frac{\partial \boldsymbol{x}}{\partial t} + \boldsymbol{\nabla}_{\mathbf{n}}\, y_i\big|_{\mu_0} \cdot \frac{\partial \boldsymbol{n}}{\partial t} + \boldsymbol{\nabla}_{\mathbf{h}}\, y_i\big|_{\mu_0} \cdot \frac{\partial \boldsymbol{h}}{\partial t} \tag{131}$$

This can be re-expressed as

$$\frac{\partial \boldsymbol{y}}{\partial t} \approx \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0} \frac{\partial \boldsymbol{x}}{\partial t} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} \frac{\partial \boldsymbol{n}}{\partial t} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0} \frac{\partial \boldsymbol{h}}{\partial t} \tag{132}$$

The expected value of equation 132, across the clean speech and additive noise variables, gives the corrupted speech delta parameters

$$\boldsymbol{\mu}_{\Delta y}^{(m)} \approx \mathcal{E}\left\{ \frac{\partial \boldsymbol{y}}{\partial t} \right\} \tag{133}$$

$$\approx \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0} \boldsymbol{\mu}_{\Delta x}^{(m)} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} \boldsymbol{\mu}_{\Delta n} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0} \boldsymbol{\mu}_{\Delta h} \tag{134}$$

If it is assumed that the additive noise is stationary, hence $\boldsymbol{\mu}_{\Delta n} = 0$, and the convolutional noise invariant, implying $\boldsymbol{\mu}_{\Delta h} = 0$, then

$$\boldsymbol{\mu}_{\Delta y}^{(m)} \approx \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0} \boldsymbol{\mu}_{\Delta x}^{(m)} \tag{135}$$

The variance of equation 132 about the expansion point $\boldsymbol{\mu}_0$ is

$$\boldsymbol{\Sigma}_{\Delta y}^{(m)} \approx \mathcal{E}\left\{ \frac{\partial \boldsymbol{y}}{\partial t} \frac{\partial \boldsymbol{y}}{\partial t}^{\mathsf{T}} \right\} - \boldsymbol{\mu}_{\Delta y}^{(m)} \boldsymbol{\mu}_{\Delta y}^{(m)\mathsf{T}} \tag{136}$$

$$\approx \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0} \boldsymbol{\Sigma}_{\Delta x}^{(m)} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0}^{\mathsf{T}} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} \boldsymbol{\Sigma}_{\Delta n} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}^{\mathsf{T}} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0} \boldsymbol{\Sigma}_{\Delta h} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0}^{\mathsf{T}} \tag{137}$$

The assumption of channel invariance translates to zero channel variance, hence

$$\boldsymbol{\Sigma}_{\Delta y}^{(m)} \approx \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0} \boldsymbol{\Sigma}_{\Delta x}^{(m)} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0}^{\mathsf{T}} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} \boldsymbol{\Sigma}_{\Delta n} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}^{\mathsf{T}} \tag{138}$$

A.1 Acceleration Coefficients

The second time derivatives are also typically used in standard recognisers. Differentiating equation 132 with respect to time, gives the equivalent form found in [1], however embeds the VTS approximation in the partial derivative. If we start from equation 127, while again assuming invariant convolutional noise, then

$$\frac{\partial^2 y_i}{\partial t^2} = \frac{\partial}{\partial t}\left\{\frac{\partial y_i}{\partial t}\right\} \tag{139}$$

$$= \frac{\partial}{\partial t}\left\{\boldsymbol{\nabla}_{\mathbf{x}}\, y_i \cdot \frac{\partial \boldsymbol{x}}{\partial t} + \boldsymbol{\nabla}_{\mathbf{n}}\, y_i \cdot \frac{\partial \boldsymbol{n}}{\partial t}\right\} \tag{140}$$

$$= \frac{\partial \boldsymbol{\nabla}_{\mathbf{x}}\, y_i}{\partial t} \cdot \frac{\partial \boldsymbol{x}}{\partial t} + \boldsymbol{\nabla}_{\mathbf{x}}\, y_i \cdot \frac{\partial^2 \boldsymbol{x}}{\partial t^2} + \frac{\partial \boldsymbol{\nabla}_{\mathbf{n}}\, y_i}{\partial t} \cdot \frac{\partial \boldsymbol{n}}{\partial t} + \boldsymbol{\nabla}_{\mathbf{n}}\, y_i \cdot \frac{\partial^2 \boldsymbol{n}}{\partial t^2} \tag{141}$$

$$= \left\{\frac{\partial^2 y_i}{\partial \boldsymbol{x} \partial \boldsymbol{x}} \frac{\partial \boldsymbol{x}}{\partial t} + \frac{\partial^2 y_i}{\partial \boldsymbol{n} \partial \boldsymbol{x}} \frac{\partial \boldsymbol{n}}{\partial t}\right\} \cdot \frac{\partial \boldsymbol{x}}{\partial t} + \boldsymbol{\nabla}_{\mathbf{x}}\, y_i \cdot \frac{\partial^2 \boldsymbol{x}}{\partial t^2} + \left\{\frac{\partial^2 y_i}{\partial \boldsymbol{x} \partial \boldsymbol{n}} \frac{\partial \boldsymbol{x}}{\partial t} + \frac{\partial^2 y_i}{\partial \boldsymbol{n} \partial \boldsymbol{n}} \frac{\partial \boldsymbol{n}}{\partial t}\right\} \cdot \frac{\partial \boldsymbol{n}}{\partial t} + \boldsymbol{\nabla}_{\mathbf{n}}\, y_i \cdot \frac{\partial^2 \boldsymbol{n}}{\partial t^2} \tag{142}$$

Since there are no cross terms in a VTS approximation of the clean speech, the cross partial derivatives are null, simplifying equation 142 to

$$\frac{\partial^2 y_i}{\partial t^2} \approx \frac{\partial^2 y_i}{\partial \boldsymbol{x} \partial \boldsymbol{x}} \frac{\partial \boldsymbol{x}}{\partial t} \cdot \frac{\partial \boldsymbol{x}}{\partial t} + \boldsymbol{\nabla}_{\mathbf{x}}\, y_i \cdot \frac{\partial^2 \boldsymbol{x}}{\partial t^2} + \frac{\partial^2 y_i}{\partial \boldsymbol{n} \partial \boldsymbol{n}} \frac{\partial \boldsymbol{n}}{\partial t} \cdot \frac{\partial \boldsymbol{n}}{\partial t} + \boldsymbol{\nabla}_{\mathbf{n}}\, y_i \cdot \frac{\partial^2 \boldsymbol{n}}{\partial t^2} \tag{143}$$

Substituting a first-order VTS approximation of the corrupted speech y_i will also result in the second-order partial derivatives being null

$$\frac{\partial^2 y_i}{\partial t^2} \approx \frac{\partial^2 y_{vts,i}}{\partial t^2} = \boldsymbol{\nabla}_{\mathbf{x}}\, y_i\big|_{\mu_0} \cdot \frac{\partial^2 \boldsymbol{x}}{\partial t^2} + \boldsymbol{\nabla}_{\mathbf{n}}\, y_i\big|_{\mu_0} \cdot \frac{\partial^2 \boldsymbol{n}}{\partial t^2} + \tag{144}$$

which may also be written as

$$\frac{\partial^2 y_i}{\partial t^2} \approx \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0} \frac{\partial^2 \boldsymbol{x}}{\partial t^2} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} \frac{\partial^2 \boldsymbol{n}}{\partial t^2} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0} \frac{\partial^2 \boldsymbol{h}}{\partial t^2} \tag{145}$$

Since equation 145 is similar in form to equation 132, thus it is obvious the first and second moments of the acceleration coefficients of the corrupted speech distribution are also similar

$$\boldsymbol{\mu}_{\Delta^2 y}^{(m)} \approx \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0} \boldsymbol{\mu}_{\Delta^2 x}^{(m)} \tag{146}$$

$$\boldsymbol{\Sigma}_{\Delta^2 y}^{(m)} \approx \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0} \boldsymbol{\Sigma}_{\Delta^2 x}^{(m)} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0}^{\mathsf{T}} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} \boldsymbol{\Sigma}_{\Delta^2 n} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}^{\mathsf{T}} \tag{147}$$

This is the same result as found in [1], however investigating higher order VTS approximations for the acceleration coefficients may be fruitful.

Alternatively, a second order VTS approximation of the corrupted speech may be made to give a better estimation of the acceleration coefficients. This is

$$y_{2vts,i} = y_i\big|_{\mu_0} + \boldsymbol{\nabla}_{\mathbf{x}}\, y_i\big|_{\mu_0} \cdot \left(\boldsymbol{x} - \boldsymbol{\mu}_x^{(m)}\right) + \boldsymbol{\nabla}_{\mathbf{n}}\, y_i\big|_{\mu_0} \cdot (\boldsymbol{n} - \boldsymbol{\mu}_n) + \boldsymbol{\nabla}_{\mathbf{h}}\, y_i\big|_{\mu_0} \cdot (\boldsymbol{h} - \boldsymbol{\mu}_h) + \tag{148}$$

$$\frac{1}{2} \frac{\partial^2 y_i}{\partial \boldsymbol{x} \partial \boldsymbol{x}} \cdot \left(\boldsymbol{x} - \boldsymbol{\mu}_x^{(m)}\right) \cdot \left(\boldsymbol{x} - \boldsymbol{\mu}_x^{(m)}\right) + \frac{1}{2} \frac{\partial^2 y_i}{\partial \boldsymbol{n} \partial \boldsymbol{n}} \cdot (\boldsymbol{n} - \boldsymbol{\mu}_n) \cdot (\boldsymbol{n} - \boldsymbol{\mu}_n) + \frac{1}{2} \frac{\partial^2 y_i}{\partial \boldsymbol{h} \partial \boldsymbol{h}} \cdot (\boldsymbol{h} - \boldsymbol{\mu}_h) \cdot (\boldsymbol{h} - \boldsymbol{\mu}_h)$$

The second order partial derivative matrices have elements

$$\frac{\partial^2 y_i}{\partial x_j \partial x_k} = \frac{\partial^2 y_i}{\partial n_j \partial n_k} = \frac{\partial^2 y_i}{\partial h_j \partial h_k} \tag{149}$$

$$= \sum_{i=1}^{D_s} c_{id} \left(\frac{\exp(\boldsymbol{c}_d^{-1}(\boldsymbol{\mu}_n - \boldsymbol{\mu}_x^{(m)} - \boldsymbol{\mu}_h))}{1 + \exp(\boldsymbol{c}_d^{-1}(\boldsymbol{\mu}_n - \boldsymbol{\mu}_x^{(m)} - \boldsymbol{\mu}_h))}\right) \left(\frac{1}{1 + \exp(\boldsymbol{c}_d^{-1}(\boldsymbol{\mu}_n - \boldsymbol{\mu}_x^{(m)} - \boldsymbol{\mu}_h))}\right) c_{dj}^{-1} c_{dk}^{-1} \tag{150}$$

for row j and column k.

B Derivative of Auxiliary wrt Static Noise Means

To find new estimates of the additive and convolutional noise, the partial derivatives of the auxiliary function in equation 51 with respect to these two vectors is necessary. Differentiating the auxiliary function

$$\frac{\partial}{\partial \hat{\boldsymbol{\mu}}_n} \mathcal{Q}(\boldsymbol{\Phi}; \hat{\boldsymbol{\Phi}}_{\hat{\mu}}) = \sum_{t=1}^{T} \sum_{m=1}^{M} \gamma_t^{(m)} \frac{\partial}{\partial \hat{\boldsymbol{\mu}}_n} \left[\log p(\boldsymbol{o}_t | \hat{\boldsymbol{\Phi}}_{\hat{\mu}}, \mathcal{M}, m) \right] \tag{151}$$

By using the first-order VTS approximation for compensation, only the static portion of the auxiliary function is dependent on the static additive and convolutional noise

$$\frac{\partial}{\partial \hat{\boldsymbol{\mu}}_n} \mathcal{Q}(\boldsymbol{\Phi}; \hat{\boldsymbol{\Phi}}_{\hat{\mu}}) = \sum_{t=1}^{T} \sum_{m=1}^{M} \gamma_t^{(m)} \frac{\partial}{\partial \hat{\boldsymbol{\mu}}_n} \left[-\frac{1}{2} \left(\boldsymbol{y}_t - \hat{\boldsymbol{\mu}}_y^{(m)} \right)^{\mathsf{T}} \boldsymbol{\Sigma}_y^{(m)\text{-}1} \left(\boldsymbol{y}_t - \hat{\boldsymbol{\mu}}_y^{(m)} \right) \right] \tag{152}$$

$$= \sum_{t=1}^{T} \sum_{m=1}^{M} \gamma_t^{(m)} \frac{\partial}{\partial \hat{\boldsymbol{\mu}}_n} \left[-\frac{1}{2} \left(\boldsymbol{y}_t^{\mathsf{T}} \boldsymbol{\Sigma}_y^{(m)\text{-}1} \boldsymbol{y}_t - 2 \hat{\boldsymbol{\mu}}_y^{(m)\mathsf{T}} \boldsymbol{\Sigma}_y^{(m)\text{-}1} \boldsymbol{y}_t + \hat{\boldsymbol{\mu}}_y^{(m)\mathsf{T}} \boldsymbol{\Sigma}_y^{(m)\text{-}1} \hat{\boldsymbol{\mu}}_y^{(m)} \right) \right] \tag{153}$$

The derivatives of the three terms, whilst using the mean of the first-order approximation of the corrupted speech from equation 59, are

$$\frac{\partial}{\partial \hat{\boldsymbol{\mu}}_n} \boldsymbol{y}_t^{\mathsf{T}} \boldsymbol{\Sigma}_y^{(m)\text{-}1} \boldsymbol{y}_t = 0 \tag{154}$$

$$\frac{\partial}{\partial \hat{\boldsymbol{\mu}}_n} \hat{\boldsymbol{\mu}}_y^{(m)\mathsf{T}} \boldsymbol{\Sigma}_y^{(m)\text{-}1} \boldsymbol{y}_t = \frac{\partial}{\partial \hat{\boldsymbol{\mu}}_n} \left[\left(\boldsymbol{\mu}_y^{(m)} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} (\hat{\boldsymbol{\mu}}_n - \boldsymbol{\mu}_n) + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0} (\hat{\boldsymbol{\mu}}_h - \boldsymbol{\mu}_h) \right)^{\mathsf{T}} \boldsymbol{\Sigma}_y^{(m)\text{-}1} \boldsymbol{y}_t \right] \tag{155}$$

$$= \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}^{\mathsf{T}} \boldsymbol{\Sigma}_y^{(m)\text{-}1} \boldsymbol{y}_t \tag{156}$$

$$\frac{\partial}{\partial \hat{\boldsymbol{\mu}}_n} \hat{\boldsymbol{\mu}}_y^{(m)\mathsf{T}} \boldsymbol{\Sigma}_y^{(m)\text{-}1} \hat{\boldsymbol{\mu}}_y^{(m)} = \frac{\partial}{\partial \hat{\boldsymbol{\mu}}_n} \left[\left(\boldsymbol{\mu}_y^{(m)} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} (\hat{\boldsymbol{\mu}}_n - \boldsymbol{\mu}_n) + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0} (\hat{\boldsymbol{\mu}}_h - \boldsymbol{\mu}_h) \right)^{\mathsf{T}} \boldsymbol{\Sigma}_y^{(m)\text{-}1} \times \right. \tag{157}$$

$$\left. \left(\boldsymbol{\mu}_y^{(m)} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} (\hat{\boldsymbol{\mu}}_n - \boldsymbol{\mu}_n) + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0} (\hat{\boldsymbol{\mu}}_h - \boldsymbol{\mu}_h) \right) \right]$$

$$= 2 \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}^{\mathsf{T}} \boldsymbol{\Sigma}_y^{(m)\text{-}1} \hat{\boldsymbol{\mu}}_y^{(m)} \tag{158}$$

Substituting these results gives

$$\frac{\partial}{\partial \hat{\boldsymbol{\mu}}_n} \mathcal{Q}(\boldsymbol{\Phi}; \hat{\boldsymbol{\Phi}}_{\hat{\mu}}) = \sum_{t=1}^{T} \sum_{m=1}^{M} \gamma_t^{(m)} \left[0 + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}^{\mathsf{T}} \boldsymbol{\Sigma}_y^{(m)\text{-}1} \boldsymbol{y}_t - \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}^{\mathsf{T}} \boldsymbol{\Sigma}_y^{(m)\text{-}1} \hat{\boldsymbol{\mu}}_y^{(m)} \right] \tag{159}$$

$$= \sum_{t=1}^{T} \sum_{m=1}^{M} \gamma_t^{(m)} \left[\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}^{\mathsf{T}} \boldsymbol{\Sigma}_y^{(m)\text{-}1} (\boldsymbol{y}_t - \hat{\boldsymbol{\mu}}_y^{(m)}) \right] \tag{160}$$

$$= \sum_{t=1}^{T} \sum_{m=1}^{M} \gamma_t^{(m)} \times \tag{161}$$

$$\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}^{\mathsf{T}} \boldsymbol{\Sigma}_y^{(m)\text{-}1} \left(\boldsymbol{y}_t - \left(\boldsymbol{\mu}_y^{(m)} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} (\hat{\boldsymbol{\mu}}_n - \boldsymbol{\mu}_n) + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0} (\hat{\boldsymbol{\mu}}_h - \boldsymbol{\mu}_h) \right) \right) = 0$$

and similarly it can be shown that

$$\frac{\partial}{\partial \hat{\boldsymbol{\mu}}_h}\mathcal{Q}(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}_{\hat{\mu}}) = \sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_t^{(m)}\left[\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0}^{\mathsf{T}}\boldsymbol{\Sigma}_y^{(m)\text{-}1}(\boldsymbol{y}_t-\hat{\boldsymbol{\mu}}_y^{(m)})\right] \tag{162}$$

$$= \sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_t^{(m)}\times \tag{163}$$

$$\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0}^{\mathsf{T}}\boldsymbol{\Sigma}_y^{(m)\text{-}1}\left(\boldsymbol{y}_t-\left(\boldsymbol{\mu}_y^{(m)}+\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}(\hat{\boldsymbol{\mu}}_n-\boldsymbol{\mu}_n)+\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0}(\hat{\boldsymbol{\mu}}_h-\boldsymbol{\mu}_h)\right)\right)=0$$

To solve for $\hat{\boldsymbol{\mu}}_n$ and $\hat{\boldsymbol{\mu}}_h$, the system of equations (161, 163) can be expressed as

$$\sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_t^{(m)}\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}^{\mathsf{T}}\boldsymbol{\Sigma}_y^{(m)\text{-}1}\left(\boldsymbol{y}_t-\boldsymbol{\mu}_y^{(m)}+\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}\boldsymbol{\mu}_n+\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0}\boldsymbol{\mu}_h-\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}\hat{\boldsymbol{\mu}}_n-\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0}\hat{\boldsymbol{\mu}}_h\right)=$$

$$\boldsymbol{d}-\boldsymbol{E}\hat{\boldsymbol{\mu}}_h-\boldsymbol{F}\hat{\boldsymbol{\mu}}_n=0 \tag{164}$$

$$\sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_t^{(m)}\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0}^{\mathsf{T}}\boldsymbol{\Sigma}_y^{(m)\text{-}1}\left(\boldsymbol{y}_t-\boldsymbol{\mu}_y^{(m)}+\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}\boldsymbol{\mu}_n+\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0}\boldsymbol{\mu}_h-\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}\hat{\boldsymbol{\mu}}_n-\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}}\Big|_{\mu_0}\hat{\boldsymbol{\mu}}_h\right)=$$

$$\boldsymbol{g}-\boldsymbol{H}\hat{\boldsymbol{\mu}}_h-\boldsymbol{J}\hat{\boldsymbol{\mu}}_n=0 \tag{165}$$

C Estimating the Additive Noise Variance

The gradient of the auxiliary function is useful for determining the optimal additive noise variance in a ML fashion. For the static additive noise variance this is

$$\frac{\partial}{\partial \hat{\boldsymbol{\Sigma}}_n}\mathcal{Q}(\boldsymbol{\Phi};\hat{\boldsymbol{\Phi}}) = \sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_t^{(m)}\frac{\partial}{\partial \hat{\boldsymbol{\Sigma}}_n}\Big[\log p\Big(\boldsymbol{o}_t\big|\hat{\boldsymbol{\Phi}},\mathcal{M},m\Big)\Big] \tag{166}$$

$$= \sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_t^{(m)}\frac{\partial}{\partial \hat{\boldsymbol{\Sigma}}_n}\left[-\frac{1}{2}\log\big|\hat{\boldsymbol{\Sigma}}_y^{(m)}\big|-\frac{1}{2}\Big(\boldsymbol{y}_t-\boldsymbol{\mu}_y^{(m)}\Big)^{\mathsf{T}}\hat{\boldsymbol{\Sigma}}_y^{(m)\text{-}1}\Big(\boldsymbol{y}_t-\boldsymbol{\mu}_y^{(m)}\Big)\right]$$

$$= -\frac{1}{2}\sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_t^{(m)}\frac{\partial}{\partial \hat{\boldsymbol{\Sigma}}_n}\left[\log\big|\hat{\boldsymbol{\Sigma}}_y^{(m)}\big|+\Big(\boldsymbol{y}_t-\boldsymbol{\mu}_y^{(m)}\Big)^{\mathsf{T}}\hat{\boldsymbol{\Sigma}}_y^{(m)\text{-}1}\Big(\boldsymbol{y}_t-\boldsymbol{\mu}_y^{(m)}\Big)\right] \tag{167}$$

Here the gradient with the respect to the static additive noise variance is only a function of the static parameters. The two terms that are being differentiated can be examined separately and on a per dimension basis. First determine the derivative of the normalising determinant

$$\frac{\partial}{\partial \hat{\sigma}_{n,i}^2}\log\big|\hat{\boldsymbol{\Sigma}}_y^{(m)}\big| = \frac{1}{\big|\hat{\boldsymbol{\Sigma}}_y^{(m)}\big|}\frac{\partial\big|\hat{\boldsymbol{\Sigma}}_y^{(m)}\big|}{\partial \hat{\sigma}_{n,i}^2} \tag{168}$$

$$= \frac{1}{\big|\hat{\boldsymbol{\Sigma}}_y^{(m)}\big|}\big|\hat{\boldsymbol{\Sigma}}_y^{(m)}\big|\mathtt{Tr}\left[\hat{\boldsymbol{\Sigma}}_y^{(m)\text{-}1}\frac{\partial\hat{\boldsymbol{\Sigma}}_y^{(m)}}{\partial \hat{\sigma}_{n,i}^2}\right] \tag{169}$$

after using the fact that $\frac{\partial|\mathbf{A}|}{\partial x} = \big|\boldsymbol{A}\big|\mathtt{Tr}\big[\boldsymbol{A}^{-1}\frac{\partial\mathbf{A}}{\partial x}\big]$. $\mathtt{Tr}$ indicates the `trace` function. The partial derivative of the variance of the corrupted speech with respect to the variance of the additive noise is needed. The variance of the first-order VTS approximation of the corrupted speech in

equation 13 provides the relationship. Hence,

$$\frac{\partial \hat{\boldsymbol{\Sigma}}_y^{(m)}}{\partial \hat{\sigma}_{n,i}^2} = \frac{\partial}{\partial \hat{\sigma}_{n,i}^2} \left\{ \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0} \boldsymbol{\Sigma}_x^{(m)} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\mu_0}^{\mathsf{T}} + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} \hat{\boldsymbol{\Sigma}}_n \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}^{\mathsf{T}} \right\} \tag{170}$$

$$= 0 + \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} \frac{\partial \hat{\boldsymbol{\Sigma}}_n}{\partial \hat{\sigma}_{n,i}^2} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}^{\mathsf{T}} \tag{171}$$

$$= \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} \boldsymbol{\Delta}_{ii} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}^{\mathsf{T}} \tag{172}$$

$$= \left(\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}^{\mathsf{T}} \right)_i \left(\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} \right)_i \tag{173}$$

Here the D_s-square matrix $\boldsymbol{\Delta}_{m,n}$ is an all zero matrix save for a single entry of 1 at row m, column n. The notation $\left(\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\big|_{\mu_0}^{\mathsf{T}} \right)_i$ gives the i^{th} column of the Jacobian matrix $\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\big|_{\mu_0}$. Substituting this result into equation 169 gives

$$\frac{\partial}{\partial \hat{\sigma}_{n,i}^2} \log\left|\hat{\boldsymbol{\Sigma}}_y^{(m)}\right| = \mathtt{Tr}\left[\hat{\boldsymbol{\Sigma}}_y^{(m)\text{-}1} \left(\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}^{\mathsf{T}} \right)_i \left(\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} \right)_i \right] \tag{174}$$

If it is assumed the inverse corrupted speech variance is diagonal, and since the `trace` function only takes into account the diagonal terms, then this can be written as

$$\frac{\partial}{\partial \hat{\sigma}_{n,i}^2} \log\left|\hat{\boldsymbol{\Sigma}}_y^{(m)}\right| = \mathtt{Tr}\left[\hat{\boldsymbol{\Sigma}}_y^{(m)\text{-}1} \mathtt{diag}\left\{ \left(\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}^{\mathsf{T}} \right)_i \circ \left(\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}^{\mathsf{T}} \right)_i \right\} \right] \tag{175}$$

$$= \sum_{d=1}^{D_s} \frac{1}{\hat{\sigma}_{y,d}^{(m)2}} \left(\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} \right)_{d,i}^2 \tag{176}$$

Next, the derivative of the main probability term is

$$\frac{\partial}{\partial \hat{\sigma}_{n,i}^2} \left(\boldsymbol{y}_t - \boldsymbol{\mu}_y^{(m)} \right)^{\mathsf{T}} \hat{\boldsymbol{\Sigma}}_y^{(m)\text{-}1} \left(\boldsymbol{y}_t - \boldsymbol{\mu}_y^{(m)} \right) = \left(\boldsymbol{y}_t - \boldsymbol{\mu}_y^{(m)} \right)^{\mathsf{T}} \frac{\partial \hat{\boldsymbol{\Sigma}}_y^{(m)\text{-}1}}{\partial \hat{\sigma}_{n,i}^2} \left(\boldsymbol{y}_t - \boldsymbol{\mu}_y^{(m)} \right) \tag{177}$$

$$= \left(\boldsymbol{y}_t - \boldsymbol{\mu}_y^{(m)} \right)^{\mathsf{T}} \left(-\hat{\boldsymbol{\Sigma}}_y^{(m)\text{-}1} \frac{\partial \hat{\boldsymbol{\Sigma}}_y^{(m)}}{\partial \hat{\sigma}_{n,i}^2} \hat{\boldsymbol{\Sigma}}_y^{(m)\text{-}1} \right) \left(\boldsymbol{y}_t - \boldsymbol{\mu}_y^{(m)} \right) \tag{178}$$

after again applying the identity $\frac{\partial \mathbf{A}^{\text{-}1}}{\partial x} = -\boldsymbol{A}^{\text{-}1} \frac{\partial \mathbf{A}}{\partial x} \boldsymbol{A}^{\text{-}1}$. Substituting the result from equation 172 yields

$$\frac{\partial}{\partial \hat{\sigma}_{n,i}^2} \left(\boldsymbol{y}_t - \boldsymbol{\mu}_y^{(m)} \right)^{\mathsf{T}} \hat{\boldsymbol{\Sigma}}_y^{(m)\text{-}1} \left(\boldsymbol{y}_t - \boldsymbol{\mu}_y^{(m)} \right) = -\left(\boldsymbol{y}_t - \boldsymbol{\mu}_y^{(m)} \right)^{\mathsf{T}} \left(\hat{\boldsymbol{\Sigma}}_y^{(m)\text{-}1} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} \boldsymbol{\Delta}_{ii} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}^{\mathsf{T}} \hat{\boldsymbol{\Sigma}}_y^{(m)\text{-}1} \right) \left(\boldsymbol{y}_t - \boldsymbol{\mu}_y^{(m)} \right) \tag{179}$$

$$= -\left[\left(\boldsymbol{y}_t - \boldsymbol{\mu}_y^{(m)} \right)^{\mathsf{T}} \hat{\boldsymbol{\Sigma}}_y^{(m)\text{-}1} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0} \right]_i^2 \tag{180}$$

This is the square of the i^{th} element of the row vector, and when $\hat{\boldsymbol{\Sigma}}_y^{(m)}$ is diagonalised, can also be written as

$$\frac{\partial}{\partial \hat{\sigma}_{n,i}^2} \left(\boldsymbol{y}_t - \boldsymbol{\mu}_y^{(m)} \right)^{\mathsf{T}} \hat{\boldsymbol{\Sigma}}_y^{(m)\text{-}1} \left(\boldsymbol{y}_t - \boldsymbol{\mu}_y^{(m)} \right) = -\left[\sum_{d=1}^{D_s} \frac{y_{t,d} - \mu_{y,d}^{(m)}}{\hat{\sigma}_{y,d}^{(m)2}} \left(\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{n}}\Big|_{\mu_0}^{\mathsf{T}} \right)_d \right]_i^2 \tag{181}$$

and ignoring the cross terms, as is the case when the covariance matrix is diagonalised

$$\frac{\partial}{\partial\hat{\sigma}^2_{n,i}}\Big(\boldsymbol{y}_t-\boldsymbol{\mu}_y^{(m)}\Big)^{\mathsf{T}}\hat{\boldsymbol{\Sigma}}_y^{(m)\text{-}1}\Big(\boldsymbol{y}_t-\boldsymbol{\mu}_y^{(m)}\Big)=-\sum_{d=1}^{D_s}\left(\frac{y_{t,d}-\mu_{y,d}^{(m)}}{\hat{\sigma}_{y,d}^{(m)2}}\right)^2\left(\frac{\partial\boldsymbol{y}}{\partial\boldsymbol{n}}\Big|_{\mu_0}^{\mathsf{T}}\right)^2_{d,i} \tag{182}$$

Hence, the gradient of the auxiliary function with respect to the static noise variances from equation 78 simplifies to

$$\begin{aligned}
\frac{\partial}{\partial\hat{\sigma}^2_{n,i}}\mathcal{Q}(\boldsymbol{\Phi},\hat{\boldsymbol{\Phi}}) &= -\frac{1}{2}\sum_{m=1}^{M}\sum_{t=1}^{T}\gamma_t^{(m)}\left[\sum_{d=1}^{D_s}\frac{1}{\hat{\sigma}_{y,d}^{(m)2}}\left(\frac{\partial\boldsymbol{y}}{\partial\boldsymbol{n}}\Big|_{\mu_0}\right)^2_{d,i}-\sum_{d=1}^{D_s}\left(\frac{y_{t,d}-\mu_{y,d}^{(m)}}{\hat{\sigma}_{y,d}^{(m)2}}\right)^2\left(\frac{\partial\boldsymbol{y}}{\partial\boldsymbol{n}}\Big|_{\mu_0}\right)^2_{d,i}\right] && (183)\\
&= -\frac{1}{2}\sum_{m=1}^{M}\sum_{t=1}^{T}\gamma_t^{(m)}\left[\sum_{d=1}^{D_s}\left(\frac{\partial\boldsymbol{y}}{\partial\boldsymbol{n}}\Big|_{\mu_0}\right)^2_{d,i}\left\{\frac{1}{\hat{\sigma}_{y,d}^{(m)2}}-\left(\frac{y_{t,d}-\mu_{y,d}^{(m)}}{\hat{\sigma}_{y,d}^{(m)2}}\right)^2\right\}\right] && (184)\\
&= -\frac{1}{2}\sum_{m=1}^{M}\sum_{d=1}^{D_s}\left(\frac{\partial\boldsymbol{y}}{\partial\boldsymbol{n}}\Big|_{\mu_0}\right)^2_{d,i}\sum_{t=1}^{T}\frac{\gamma_t^{(m)}}{\hat{\sigma}_{y,d}^{(m)2}}\left\{1-\left(\frac{y_{t,d}-\mu_{y,d}^{(m)}}{\hat{\sigma}_{y,d}^{(m)}}\right)^2\right\} && (185)\\
&= -\frac{1}{2}\sum_{m=1}^{M}\sum_{d=1}^{D_s}\left(\frac{\partial\boldsymbol{y}}{\partial\boldsymbol{n}}\Big|_{\mu_0}\right)^2_{d,i}\sum_{t=1}^{T}\frac{\gamma_t^{(m)}}{\hat{\sigma}_{y,d}^{(m)2}}\left\{1-\frac{y_{t,d}^2-2y_{t,d}\mu_{y,d}^{(m)}+\mu_{y,d}^{(m)2}}{\hat{\sigma}_{y,d}^{(m)2}}\right\} && (186)\\
&= -\frac{1}{2}\sum_{m=1}^{M}\sum_{d=1}^{D_s}\left(\frac{\partial\boldsymbol{y}}{\partial\boldsymbol{n}}\Big|_{\mu_0}\right)^2_{d,i}\frac{1}{\hat{\sigma}_{y,d}^{(m)2}}\left\{L^{(m)}-\frac{p_d^{(m)}-2q_d^{(m)}\mu_{y,d}^{(m)}+L^{(m)}\mu_{y,d}^{(m)2}}{\hat{\sigma}_{y,d}^{(m)2}}\right\} && (187)\\
&= -\frac{1}{2}\sum_{m=1}^{M}\sum_{d=1}^{D_s}\left(\frac{\partial\boldsymbol{y}}{\partial\boldsymbol{n}}\Big|_{\mu_0}\right)^2_{d,i}\frac{1}{\hat{\sigma}_{y,d}^{(m)2}}\left\{\left(1-\frac{\mu_{y,d}^{(m)2}}{\hat{\sigma}_{y,d}^{(m)2}}\right)L^{(m)}-\frac{p_d^{(m)}-2q_d^{(m)}\mu_{y,d}^{(m)}}{\hat{\sigma}_{y,d}^{(m)2}}\right\} && (188)
\end{aligned}$$

where the sufficient statistics $\boldsymbol{p}^{(m)}$ and $\boldsymbol{q}^{(m)}$ are defined as

$$p_d^{(m)}=\sum_{t=1}^{T}\gamma_t^{(m)}y_{t,d}^2 \qquad\qquad q_d^{(m)}=\sum_{t=1}^{T}\gamma_t^{(m)}y_{t,d} \tag{189}$$