CAMBRIDGE UNIVERSITY

ENGINEERING DEPARTMENT

Issues with Uncertainty Decoding for Noise Robust Automatic Speech Recognition

H. Liao and M.J.F. Gales

CUED/F-INFENG/TR.549
August 4, 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

Interest is growing in a class of robustness algorithms that exploit the notion of uncertainty introduced by environmental noise. The majority of these techniques share the property that the uncertainty of an observation due to noise is propagated to the recogniser, resulting in increased model variances. Using appropriate approximations, efficient implementations may be obtained, with the goal of achieving near model-based performance without the associated computational cost. Unfortunately, uncertainty decoding forms that compute the uncertainty in the front-end and pass this to the decoder may suffer from a theoretical problem in low signal-to-noise ratio conditions. This report discusses how this fundamental issue arises, and demonstrates it through two schemes: `SPLICE` with uncertainty and front-end `Joint` uncertainty decoding. A method to mitigate this in the `Joint` form is presented, as well as how `SPLICE` implicitly addresses it. However, it is shown that a model-based `Joint` uncertainty decoding approach does not suffer from this limitation, like these front-end forms do, and is also competitive computationally. The issues described and performance of the various schemes are examined on two artificially corrupted corpora: AURORA 2.0 digit recognition database and the thousand-word Resource Management task.

1 Introduction

Improving the robustness of state-of-the-art speech recognisers continues to be an important research area. Current technology based on hidden Markov models using Gaussian output distributions performs well on uncorrupted speech, but in practice is very susceptible to environmental noise. Techniques such as Parallel Model Combination (PMC) [11], vector Taylor series-based compensation [17, 1] and more recently `ALGONQUIN` [18] have been shown to be very effective forms of noise compensation. Unfortunately, these are quite computationally expensive and generally intractable for real-time large vocabulary systems compared to classic front-end enhancement techniques like spectral subtraction and cepstral mean normalisation which are also less powerful.

Recently, research has focused on extending feature-based schemes by incorporating the uncertainty of the enhancement process into the recognition process. Some approaches have done so in an ad hoc fashion, by for example adding an uncertainty variance based on the position of formants [15], from a polynomial function of the signal-to-noise ratio [3] to the model variances, or the variance of the enhancement process [7, 4, 25, 8]. *Uncertainty decoding* propagates the conditional probability of the corrupted speech given the "clean" speech, $p(\boldsymbol{y}|\boldsymbol{x})$ [9, 20] into the decoding stage. Whichever the case, the uncertainty is calculated efficiently in the front-end, and passed to the recogniser as a single, simple variance offset to the recognition model components. This can provide an elegant compromise of a fast-feature based compensation scheme with model-based accuracy.

Front-end based uncertainty compensation schemes have demonstrated good results for a variety of different tasks and environments, however they suffer from some inherent flaws. One broad approach is to consider that feature enhancement leaves residual observation uncertainty; thus the observations may be augmented by an uncertainty variance which is the expected square error of the enhancement process. However, there is no mathematical basis for this "intuitive" approach. For front-end uncertainty decoding, as first described in [9], there is an even more fundamental problem. In low SNR when the noise masks the speech, the conditional distribution $p(\boldsymbol{y}|\boldsymbol{x})$ becomes independent of the clean speech $\boldsymbol{x}$. Since this distribution is determined in the front-end and marginalised against all the acoustic components, this factorisation transforms all the acoustic components to the same noise distribution. Thus no discrimination is possible, and if there are no other constraints such as a strong language model, then large numbers of insertions can take place in these areas of high uncertainty. Because model-based uncertainty decoding explicitly associates the corrupted speech conditional with the clean speech distribution, it does not suffer from this problem.

In this report, some feature enhancement schemes are described and how they can be extended to include uncertainty is presented in section 2. This extension gives a similar decoding form to the front-end uncertainty decoding method reviewed in section 3. Theoretical problem with these methods as elaborated in section 4; how it manifests in specific forms is demonstrated with the `SPLICE` and `Joint` techniques. In chapter 5, these issues and techniques are evaluated on the standard AURORA 2.0 noisy digit recognition task [14] and an artificially corrupted 1000-word Resource Management database using a NOISEX-92 sample [22]. We show that model-based uncertainty decoding does not suffer from this problem and is more effective, yet more efficient. Overall conclusions and future work directions are presented in section 6.

2 Feature enhancement

Traditionally, fast front-end noise compensation techniques, such as spectral subtraction and more recently `SPLICE` [6], have removed the noise from the observed corrupted speech $\boldsymbol{y}_t$, and passed this estimate $\hat{\boldsymbol{x}}_t$ as an if were exactly the original clean speech vector $\boldsymbol{x}_t$ to the acoustic models as shown in figure 1. Hence, given the estimate of the clean speech, the decoding likelihood is simply the evaluation of the enhanced clean speech against the uncompensated clean acoustic models

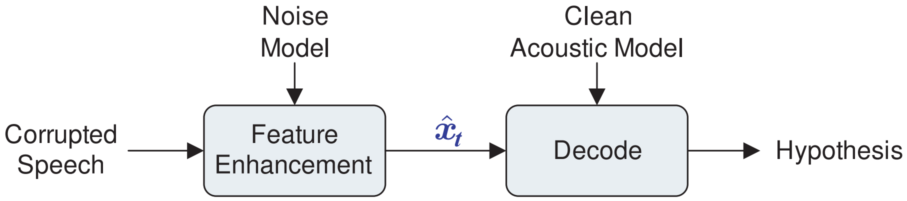

Figure 1: The standard feature enhancement process.

$$p\left(\boldsymbol{y}_t|\mathcal{M},\check{\mathcal{M}},\theta_t\right) = p\left(\hat{\boldsymbol{x}}_t|\mathcal{M},\theta_t\right) \tag{1}$$

where $\mathcal{M}$ represent the set of clean acoustic model parameters, and $\check{\mathcal{M}}$ some set of front-end parameters that may include noise or simplified speech models.

Simple MMSE enhancement schemes tend to be very efficient and take the following form for the estimate of the clean speech

$$\hat{\boldsymbol{x}}_t = \int_{\mathcal{R}^d} \boldsymbol{x}_t p\big(\boldsymbol{x}_t|\boldsymbol{y}_t,\check{\mathcal{M}}\big)\, d\boldsymbol{x}_t \tag{2}$$

$$= \mathcal{E}\left\{\boldsymbol{x}|\boldsymbol{y},\check{\mathcal{M}}\right\} \tag{3}$$

Note that the enhancement is based on $\check{\mathcal{M}}$ and is independent of the actual acoustic model parameters $\mathcal{M}$. A recent algorithm, based on this framework, is called `SPLICE`. The expected value of the clean speech given the corrupted speech is made into a piece-wise function depending on the region of the acoustic space

$$\hat{\boldsymbol{x}}_t = \sum_n \mathcal{E}\left\{\boldsymbol{x}_t|\boldsymbol{y}_t,\check{\mathcal{M}},n\right\} P(n|\boldsymbol{y},\check{\mathcal{M}}) \tag{4}$$

with the component posterior defined as

$$P\big(n|\boldsymbol{y}_t,\check{\mathcal{M}}\big) = \frac{\check{c}_n p\big(\boldsymbol{y}_t|\check{\mathcal{M}},n\big)}{\sum_{i=1}^{N} \check{c}_i p\big(\boldsymbol{y}_t|\check{\mathcal{M}},i\big)} \tag{5}$$

and the expected value of the clean speech posterior the mean of

$$p(\boldsymbol{x}_t|\boldsymbol{y}_t,n) = \mathcal{N}\big(\boldsymbol{x}_t;\boldsymbol{y}_t+\check{\boldsymbol{\mu}}^{(n)},\check{\boldsymbol{\Sigma}}^{(n)}\big) \tag{6}$$

that is

$$\mathcal{E}\left\{\boldsymbol{x}_t|\boldsymbol{y}_t,\check{\mathcal{M}},n\right\} = \boldsymbol{y}_t + \check{\boldsymbol{\mu}}^{(n)} \tag{7}$$

Often the `SPLICE` form in equation 4 is optimised by only applying the bias associated with the most likely component n^* of the corrupted speech Gaussian mixture model (GMM)

$$\hat{\boldsymbol{x}}_t = \mathcal{E}\left\{\boldsymbol{x}_t|\boldsymbol{y}_t,n^*\right\} \tag{8}$$

$$= \boldsymbol{y}_t + \check{\boldsymbol{\mu}}^{(n^*)} \tag{9}$$

Here, the corrupted speech is updated simply by the bias $\check{\boldsymbol{\mu}}^{(n^*)}$, which is the expected value of difference between the clean and corrupted speech, associated with the most probable region of acoustic space n^*. This is then used during decoding as representative of the clean speech feature vector. The corrupted acoustic space GMM with N components is given by

$$p\big(\boldsymbol{y}_t|\check{\mathcal{M}}\big) = \sum_{n=1}^{N} \check{c}_n \mathcal{N}\Big(\boldsymbol{y}_t; \boldsymbol{\mu}_y^{(n)}, \boldsymbol{\Sigma}_y^{(n)}\Big) \tag{10}$$

where $\check{c}_n$ is the component prior, and the most likely component is determined by

$$n^* = \arg\max_n \Big[\check{c}_n P\big(\boldsymbol{y}_t|n, \check{\mathcal{M}}\big)\Big] \tag{11}$$

The correction vectors can be estimated using stereo data in the following manner

$$\check{\boldsymbol{\mu}}^{(n)} = \mathcal{E}\left\{\boldsymbol{x}_t - \boldsymbol{y}_t|n\right\} \tag{12}$$

$$\check{\boldsymbol{\Sigma}}^{(n)} = \mathcal{E}\left\{(\boldsymbol{x}_t - \boldsymbol{y}_t)(\boldsymbol{x}_t - \boldsymbol{y}_t)^{\mathsf{T}}|n\right\} - \check{\boldsymbol{\mu}}^{(n)}\check{\boldsymbol{\mu}}^{(n)\mathsf{T}} \tag{13}$$

`SPLICE` has shown to be quite an effective compensation algorithm on the standard AURORA corpus [10]. It is also efficient, compensating the features is trivial, hence the main cost are the N Gaussian evaluations in equation 11 to choose the acoustic region and associated correction bias.

2.1 Decoding with observation uncertainty

Recently, standard enhancement schemes, which typically only update the feature vector and pass this on as the true clean speech to the decoder, have been extended to consider uncertainty of the de-noising process itself. So instead of assuming the feature cleaning process is exact, a distribution $p(\boldsymbol{x}|\boldsymbol{y}, \check{\mathcal{M}})$ is passed to the acoustic models representing the uncertainty in the compensation. This has been termed uncertain observations or uncertain observation decoding in [3]. The mean of this distribution is the estimate $\hat{\boldsymbol{x}}$, but with an associated variance that may be the expected square error of the enhancement. This scheme is depicted in figure 2 which can be compared with figure 1.

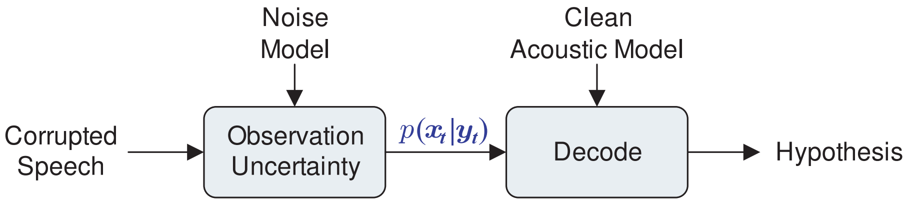

Figure 2: Feature enhancement with observation uncertainty.

Thus if the clean speech feature vector is now considered a multivariate distribution $\boldsymbol{x}_t \sim \mathcal{N}(\hat{\boldsymbol{x}}_t, \boldsymbol{\Sigma}_{\hat{\mathbf{x}}})$ the decoding likelihood requires integration over all the possible values

$$p\big(\boldsymbol{y}_t|\mathcal{M}, \hat{\mathcal{M}}, \theta_t, m\big) \approx \int_{\mathcal{R}^d} p(\boldsymbol{x}_t|\boldsymbol{y}_t, \check{\mathcal{M}}) p(\boldsymbol{x}_t|\mathcal{M}, \theta_t) d\boldsymbol{x}_t \tag{14}$$

$$= \int_{\mathcal{R}^d} \mathcal{N}(\boldsymbol{x}_t; \hat{\boldsymbol{x}}_t, \boldsymbol{\Sigma}_{\hat{\mathbf{x}}}) \mathcal{N}\Big(\boldsymbol{x}_t; \boldsymbol{\mu}^{(m)}, \boldsymbol{\Sigma}^{(m)}\Big) d\boldsymbol{x}_t \tag{15}$$

$$= \mathcal{N}\Big(\hat{\boldsymbol{x}}_t; \boldsymbol{\mu}^{(m)}, \boldsymbol{\Sigma}^{(m)} + \boldsymbol{\Sigma}_{\hat{\mathbf{x}}}\Big) \tag{16}$$

Here $\hat{\boldsymbol{x}}_t$ is again the clean speech estimate, as is normally produced from standard enhancement schemes. The variance offset $\boldsymbol{\Sigma}_{\hat{\mathbf{x}}}$ is the expected square error of this enhancement process. In `SPLICE` this is

$$\boldsymbol{\Sigma}_{\hat{\mathbf{x}}} = \check{\boldsymbol{\Sigma}}^{(n^*)} \tag{17}$$

and can be estimated from stereo data as in equation 13. Other enhancement schemes can be easily extended to provide this variance, for example considering formant frequencies as part of a heuristic measure [15], using a weighted polynomial function of the SNR in the log spectral domain [3], obtaining them from a parametric model of the clean speech [7], or from the classic Weiner filter [4].

An interesting observation uncertainty form is the model-based feature enhancement(MBFE) technique extended to account for observation uncertainty [25]. It is notable because like the front-end `Joint` uncertainty decoding form discussed in the next section, a GMM is embedded in the front-end and a joint distribution between the clean and corrupted speech is computed for each component. It differs, in that it uses the joint distribution to compute the clean speech posterior, where the clean speech estimate for a particular component n is

$$\hat{\boldsymbol{x}}_t^{(n)} = \boldsymbol{\mu}_x^{(n)} + \boldsymbol{\Sigma}_{xy}^{(n)} \left(\boldsymbol{\Sigma}_y^{(n)}\right)^{-1} \left(\boldsymbol{y}_t - \boldsymbol{\mu}_y\right) \tag{18}$$

and the associated variance or uncertainty

$$\boldsymbol{\Sigma}_{\hat{\mathbf{x}}}^{(n)} = \boldsymbol{\Sigma}_x^{(n)} - \boldsymbol{\Sigma}_{xy}^{(n)} \left(\boldsymbol{\Sigma}_y^{(n)}\right)^{-1} \boldsymbol{\Sigma}_{yx}^{(n)} \tag{19}$$

In this form of enhancement, as described in [25],the MMSE estimate is formed by summing over all the estimates, weighted by the component posterior, rather than just choosing the most likely state as in the `SPLICE` form.

With these forms of front-end uncertainty processing, the computation cost is similar to standard enhancement scheme, with small additional memory requirements for storing the uncertainty variances. However the uncertainty that is propagated for the current frame must be added to all acoustic model components. This can total in the hundreds for a small task, such as the reference AURORA recogniser [14], to the hundred of thousands in state-of-the-art recognition systems such as the CU Broadcast News system [16]. Moreover, this variance addition is not as simple to compute as say a scaling of the variances; the Gaussian normalisation term that is usually cached must also be re-calculated. As stated in [3], assuming that Gaussian evaluations comprise 50% of the total computation cost of transcribing speech, the overhead of adding uncertainty is approximately 33%. Nevertheless, applying this variance update with a single global uncertainty is far cheaper than expensive model-based techniques such as VTS compensation, PMC or `ALGONQUIN` which separately compensate each acoustic model component individually depending on the effects of the noise on that Gaussian.

While in practice good results have been obtained using observation uncertainty [4, 7, 25], there is a concern. There is no reason that the feature vector should be augmented with the expected square error of the enhancement process. Despite the presumption that enhanced observations may not be exact seems sensible, the resulting decoding form given in equation 14 does not arise from any consistent mathematical framework. Perhaps this is why the variances propagated seem ill-conditioned: they are deemed to large and imprecise [7] or reduced by a factor of ten [25]. Hence, although some robustness may have been obtained using this technique, it is fundamentally unfounded and merely a heuristic approach.

3 Uncertainty decoding

In this section, the formal uncertainty decoding framework from [9, 20] is described along with some forms that operate within this framework. A model of the corrupted speech as a function of the clean speech and noise can be expressed by a dynamic Bayesian network as shown in figure 3. Here, the noise corrupted speech observation $\boldsymbol{y}_t$ at time t is assumed to be conditionally independent of all other observations given the clean speech $\boldsymbol{x}_t$ and the noise $\boldsymbol{n}_t$ at that time. The clean speech and noise are assumed to be generated by HMMs with states θ_t^n for the noise[1] and

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

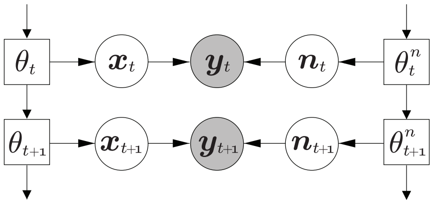

Figure 3: Uncertainty decoding DBN.

θ_t for the clean speech. Under these assumptions the likelihood of the corrupted observation is given by

$$p(\boldsymbol{y}_t|\mathcal{M},\check{\mathcal{M}},\theta_t) = \int_{\mathcal{R}^d} p(\boldsymbol{y}_t|\boldsymbol{x}_t,\check{\mathcal{M}})p(\boldsymbol{x}_t|\mathcal{M},\theta_t)d\boldsymbol{x}_t \tag{20}$$

where

$$p(\boldsymbol{y}_t|\boldsymbol{x}_t,\check{\mathcal{M}}) = \int_{\mathcal{R}^d} p(\boldsymbol{y}_t|\boldsymbol{x}_t,\boldsymbol{n}_t)p(\boldsymbol{n}_t|\check{\mathcal{M}},\theta_t^n)d\boldsymbol{n}_t \tag{21}$$

and $\check{\mathcal{M}}$ the front-end compensation model. The acoustic model $\mathcal{M}$ consists of Gaussian components each defined by a prior c_m, mean $\boldsymbol{\mu}^{(m)}$, and variance $\boldsymbol{\Sigma}^{(m)}$. The likelihood calculation thus has two distinct parts. Only the first, $p(\boldsymbol{y}_t|\boldsymbol{x}_t,\check{\mathcal{M}})$, is a function of the noise, the other is the clean speech prior which is not dependent on the noise. Hence, this marginalisation is independent of the noise given the form of $p(\boldsymbol{y}_t|\boldsymbol{x}_t,\check{\mathcal{M}})$. This uncertainty decoding framework can be depicted as in figure 4.

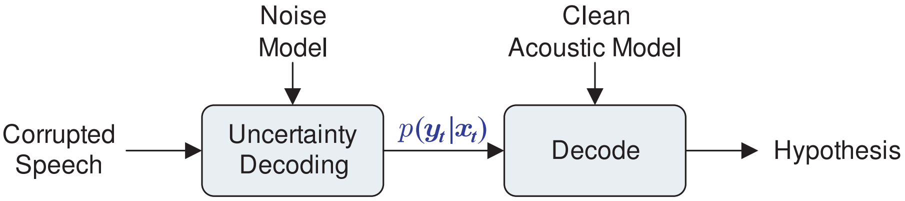

Figure 4: Uncertainty decoding framework.

The term *uncertainty decoding* can be considered to encompass forms that exploit this factorisation by determining an efficient approximation for the conditional distribution of the corrupted speech given the clean that easily completes the marginalisation and is cheap to compute. It can be decoupled from the structure of the actual acoustic models and thus there is significant freedom in choosing an appropriate form for this distribution that minimises the computational cost. If it is completely decoupled, and dependent entirely on the observed features, this gives the front-end uncertainty decoding forms. Partial decoupling, where the conditional is dependent on the "class" of the acoustic model component, yields a model-based uncertainty decoding scheme. In pure model-based approaches, such as ALGONQUIN, PMC or VTS compensation, the two distributions are fully tied by the clean speech variable. For example, ALGONQUIN explicitly computes the conditional for each model component making it impractical for large vocabulary systems.

3.1 Front-end uncertainty decoding

In front-end uncertainty decoding, a major focus is on determining a form for the conditional distribution, $p(\boldsymbol{y}_t|\boldsymbol{x}_t,\check{\mathcal{M}})$ that can be efficiently computed and independent from the back-end acoustic models. The nature of the conditional can be explored by examining the joint clean-corrupted

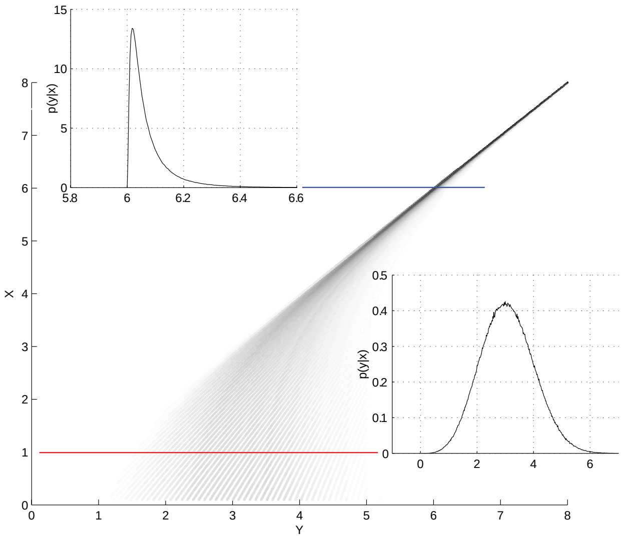

Figure 5: Joint distribution $p(x, y)$.

speech distribution as shown in figure 5. This simulation, which has also been detailed in [19] and [4], takes place in the log energy domain where x represents the clean speech and y the noise corrupted speech, where it is assumed that $y = \log(\exp(x) + \exp(n))$. This relationship is highly non-linear especially in the low SNR region to the left and clearly non-Gaussian. Nevertheless, the approach taken in uncertainty decoding is to represent the corrupted speech conditional given the clean speech with a GMM. By selecting a single, most probable component of the GMM given the observed noisy data, a single variance offset per frame is passed to the acoustic models during decoding. This is an approximation for efficiency, since not doing so would cause the complexity in the front-end process to multiple with the number of components in the back-end. The use of Gaussian distributions makes the marginalisation in equation 20 trivial. In this way, an elegant compromise is achieved with fast front-end processing providing a simple acoustic model update.

Two specific forms of front-end uncertainty decoding have been presented in the literature: `SPLICE` with uncertainty [9] and the `Joint` method [19, 20]. For both, the resultant likelihood of the corrupted speech observation using the uncertainty parameters selected from component n of the front-end GMM for state θ_t can be expressed as [20]

$$p(\boldsymbol{y}_t|\mathcal{M},\check{\mathcal{M}},\theta_t) \propto \sum_{m\in\theta_t} c_m \, \mathcal{N}\Big(\boldsymbol{A}^{(n)}\boldsymbol{y}_t+\boldsymbol{b}^{(n)};\boldsymbol{\mu}^{(m)},\boldsymbol{\Sigma}^{(m)}+\boldsymbol{\Sigma}_{\mathtt{b}}^{(n)}\Big) \tag{22}$$

where $\boldsymbol{A}^{(n)}$, $\boldsymbol{b}^{(n)}$ and $\boldsymbol{\Sigma}_{\mathtt{b}}^{(n)}$ are the compensation parameters. In form, this is exactly the same as the observation uncertainty decoding approach as given in equation 16, with

$$\hat{\boldsymbol{x}} = \boldsymbol{A}^{(n)}\boldsymbol{y}_t+\boldsymbol{b}^{(n)} \tag{23}$$

$$\boldsymbol{\Sigma}_{\hat{\mathbf{x}}} = \boldsymbol{\Sigma}_{\mathtt{b}}^{(n)} \tag{24}$$

but the parameters are derived from a fundamentally different, mathematically sound perspective. `SPLICE` with uncertainty and the `Joint` scheme give two methods for estimating such parameters using the uncertainty decoding framework.

3.1.1 `SPLICE`**with uncertainty**

`SPLICE` with uncertainty makes use of Bayes' rule to express the conditional probability of the corrupted speech given the clean speech in terms of the conditional probability of the clean speech given the corrupted speech. Though this simplifies the calculation of the distribution, an approximation for the distribution of the clean speech is required. A single Gaussian distribution is used where the global clean speech mean and variance for dimension i are $\bar{\mu}_{x,i}$ and $\bar{\sigma}^2_{x,i}$. Using this approximation, with the restriction that $\boldsymbol{A}^{(n)}$ and $\boldsymbol{\Sigma}_{\mathsf{b}}^{(n)}$ are diagonal, gives

$$a_{ii}^{(n)} = \frac{\bar{\sigma}^2_{x,i}}{\bar{\sigma}^2_{x,i} - \breve{\sigma}_i^{(n)2}} \tag{25}$$

$$b_i^{(n)} = a_{ii}^{(n)} \left(\breve{\mu}_i^{(n)} - \frac{\breve{\sigma}_i^{(n)2}}{\bar{\sigma}^2_{x,i}} \bar{\mu}_{x,i} \right) \tag{26}$$

$$\sigma_{\mathsf{b}i}^{(n)2} = a_{ii}^{(n)} \breve{\sigma}_i^{(n)2} \tag{27}$$

The parameters $\breve{\mu}_i^{(n)}$ and $\breve{\sigma}_i^{(n)2}$ are the means and variance respectively of $(x_{ti} - y_{ti})$ for the data associated with component n of the front-end corrupted speech GMM given in equation 10. In order to ensure that the uncertainty variance bias $\boldsymbol{\Sigma}_{\mathsf{b}}^{(n)}$ is positive the denominator in equation 25 is floored. In this work the floor is set to a fraction α of the global clean variance $\bar{\sigma}^2_{x,i}$. This floor effectively places a maximum value on $a_{ii}^{(n)}$ where

$$a_{ii}^{(n)} = \min \left(\frac{1}{\alpha} \;,\; \frac{\bar{\sigma}^2_{x,i}}{\bar{\sigma}^2_{x,i} - \breve{\sigma}_i^{(n)2}} \right) \tag{28}$$

The effects of this are discussed in more detail in the next section.

3.1.2 Front-end `Joint`uncertainty decoding

In the front-end version of the `Joint` uncertainty decoding scheme, the corrupted speech conditional distribution given the clean speech is modeled by a GMM. To estimate the conditional, a joint distribution of the clean and corrupted speech is estimated for each region of the acoustic space. For component n of the front-end corrupted speech GMM the joint distribution is assumed to be Gaussian with parameters

$$\begin{bmatrix} \boldsymbol{x}_t \\ \boldsymbol{y}_t \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu}_x^{(n)} \\ \boldsymbol{\mu}_y^{(n)} \end{bmatrix}, \begin{bmatrix} \boldsymbol{\Sigma}_x^{(n)} & \boldsymbol{\Sigma}_{xy}^{(n)} \\ \boldsymbol{\Sigma}_{yx}^{(n)} & \boldsymbol{\Sigma}_y^{(n)} \end{bmatrix} \right) \tag{29}$$

Given the joint distribution, the conditional distribution can be derived as follows

$$\begin{aligned} p\big(\boldsymbol{y}_t | \boldsymbol{x}_t, \check{\mathcal{M}}, n\big) &\approx \mathcal{N}\Big(\boldsymbol{y}_t; \boldsymbol{\mu}_y^{(n)} + \boldsymbol{\Sigma}_{yx}^{(n)} \boldsymbol{\Sigma}_x^{(n)\text{-}1} \big(\boldsymbol{x}_t - \boldsymbol{\mu}_x^{(n)}\big), \boldsymbol{\Sigma}_y^{(n)} - \boldsymbol{\Sigma}_{yx}^{(n)} \boldsymbol{\Sigma}_x^{(n)\text{-}1} \boldsymbol{\Sigma}_{xy}^{(n)}\Big) && (30) \\ &= \alpha^{(n)} \mathcal{N}\Big(\boldsymbol{\Sigma}_x^{(n)} \boldsymbol{\Sigma}_{yx}^{(n)\text{-}1} \big(\boldsymbol{y}_t - \boldsymbol{\mu}_y^{(n)}\big) + \boldsymbol{\mu}_x^{(n)}; \boldsymbol{x}_t, \boldsymbol{\Sigma}_x^{(n)} \boldsymbol{\Sigma}_{yx}^{(n)\text{-}1} \boldsymbol{\Sigma}_y^{(n)} \boldsymbol{\Sigma}_x^{(n)} \boldsymbol{\Sigma}_{yx}^{(n)\text{-}1} - \boldsymbol{\Sigma}_x^{(n)}\Big) && (31) \\ &= \alpha^{(n)} \mathcal{N}\Big(\boldsymbol{A}^{(n)} \boldsymbol{y}_t + \boldsymbol{b}^{(n)}; \boldsymbol{x}_t, \hat{\boldsymbol{\Sigma}}_x^{(n)}\Big) && (32) \end{aligned}$$

where $\alpha^{(n)} = \big|\boldsymbol{A}^{(n)}\big|$. This normalisation is not strictly necessary since it is the same for all likelihood calculations for each frame of speech and hence does not affect the recognition search. If instead of using the full GMM to represent the conditional, only the one associated with the most probably corrupted speech component n is used, then the compensation parameters in equation 22 are given by

$$\boldsymbol{A}^{(n)} = \boldsymbol{\Sigma}_x^{(n)} \boldsymbol{\Sigma}_{yx}^{(n)\text{-}1} \tag{33}$$

$$\boldsymbol{b}^{(n)} = \boldsymbol{\mu}_x^{(n)} - \boldsymbol{A}^{(n)} \boldsymbol{\mu}_y^{(n)} \tag{34}$$

$$\boldsymbol{\Sigma}_{\mathsf{b}}^{(n)} = \boldsymbol{A}^{(n)} \boldsymbol{\Sigma}_y^{(n)} \boldsymbol{A}^{(n)\mathsf{T}} - \boldsymbol{\Sigma}_x^{(n)} \tag{35}$$

Though the feature transform and variance bias may be full for the Joint scheme, they are typically made diagonal for the front-end form for efficiency. The selection of the appropriate conditional distribution based on the observed corrupted speech is a significant approximation and is discussed in more detail in [19].

It is interesting to compare these parameters with the ones derived using the clean speech posterior in the observation uncertainty form given by equations 18 and 19. Although both estimate joint distributions for front-end components representing acoustic regions of the corrupted speech space, and have similar decoding likelihood forms, the actual compensation parameters are completely different. For the MBFE form, these are

$$\boldsymbol{A}^{(n)} = \boldsymbol{\Sigma}_{xy}^{(n)}\boldsymbol{\Sigma}_{y}^{(n)\text{-1}} \tag{36}$$

$$\boldsymbol{b}^{(n)} = \boldsymbol{\mu}_{x}^{(n)} - \boldsymbol{A}^{(n)}\boldsymbol{\mu}_{y}^{(n)} \tag{37}$$

$$\boldsymbol{\Sigma}_{\mathsf{b}}^{(n)} = \boldsymbol{\Sigma}_{\hat{\mathbf{x}}}^{(n)} = \boldsymbol{\Sigma}_{x}^{(n)} - \boldsymbol{A}^{(n)}\boldsymbol{\Sigma}_{yx}^{(n)} \tag{38}$$

Note that normally with MBFE, the clean speech estimate is formed from a weighted contribution from all the components in the front-end GMM, not just the most likely. Thus, the MBFE form here is not exactly as originally described in [25].

3.2 Model-based uncertainty decoding

In the last section describing front-end uncertainty decoding, the conditional in equation 21 is completely decoupled from the acoustic models. However, more powerful forms can arise from maintaining this link. In the ALGONQUIN scheme, an *interaction likelihood* $\boldsymbol{\Psi}$ [18] captures the residual error in the mismatch function $f(\boldsymbol{x}_t, \boldsymbol{n}_t)$. This is propagated to the recognition search as the conditional in the uncertainty decoding framework

$$p(\boldsymbol{y}_t|\boldsymbol{x}_t) = \mathcal{N}(\boldsymbol{y}_t; f(\boldsymbol{x}_t, \boldsymbol{n}_t), \boldsymbol{\Psi}) \tag{39}$$

However, these model-based schemes are known to be computationally expensive. For example ALGONQUIN uses a variation Bayes algorithm to iteratively approximate the non-Gaussian corrupted speech distribution; this is conducted for every recognition component. Therefore, ALGONQUIN is comparable in form to pure model-based schemes such as PMC or VTS compensation as the effect of the noise is considered independently for each acoustic model component.

For model-based Joint uncertainty decoding, there is a middle ground between front-end uncertainty decoding, and pure-model based forms. The uncertainty parameters can be estimated for a group of similar acoustic model components rather than globally in the front-end or for each model component separately. Compared to front-end uncertainty decoding, instead of having each component in the front-end associated with a region of acoustic space n, link it to a set of similar recognition model components r. For example, one may choose to have two recognition classes, one for silence and another for speech; more classes may be derived by using a regression tree depending on the amount of data available [23]. The joint distribution can than be computed over this class of recognition components r where the mean vectors and covariance matrices of the clean and corrupted speech are given by

$$\boldsymbol{\mu}_{x}^{(r)} = \frac{\sum_{m\in r}\gamma_m(t)\boldsymbol{x}_t}{\sum_{m\in r}\gamma_m(t)} \qquad \boldsymbol{\mu}_{y}^{(r)} = \frac{\sum_{m\in r}\gamma_m(t)\boldsymbol{y}_t}{\sum_{m\in r}\gamma_m(t)} \tag{40}$$

$$\boldsymbol{\Sigma}_{x}^{(r)} = \frac{\sum_{m\in r}\gamma_m(t)\boldsymbol{x}_t\boldsymbol{x}_t^{\mathsf{T}}}{\sum_{m\in r}\gamma_m(t)} - \boldsymbol{\mu}_{x}^{(r)}\boldsymbol{\mu}_{x}^{(r)\mathsf{T}} \qquad \boldsymbol{\Sigma}_{y}^{(r)} = \frac{\sum_{m\in r}\gamma_m(t)\boldsymbol{y}_t\boldsymbol{y}_t^{\mathsf{T}}}{\sum_{m\in r}\gamma_m(t)} - \boldsymbol{\mu}_{y}^{(r)}\boldsymbol{\mu}_{y}^{(r)\mathsf{T}} \tag{41}$$

where $\gamma_m(t)$ is the component posterior at time instance t. The notation $\sum_{m\in r}$ denotes summation over the recognition components m in model class r. The cross-covariance terms between the clean and corrupted speech are then given by

$$\boldsymbol{\Sigma}_{xy}^{(r)} = \frac{\sum_{m\in r}\gamma_m(t)\boldsymbol{x}_t\boldsymbol{y}_t^{\mathsf{T}}}{\sum_{m\in r}\gamma_m(t)} - \boldsymbol{\mu}_{x}^{(r)}\boldsymbol{\mu}_{y}^{(r)\mathsf{T}} \tag{42}$$

Having obtained the component parameters, the compensation parameters can be derived using equations 33 to 35. Figure 6 depicts how these parameters are estimated for a class of recognition components r.

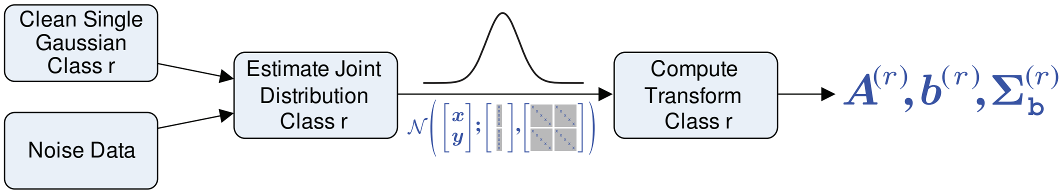
Clean Single Gaussian Class r
Noise Data
Estimate Joint Distribution Class r
$\mathcal{N}\left(\begin{bmatrix} x \\ y \end{bmatrix};\ \ldots\right)$
Compute Transform Class r
$\boldsymbol{A}^{(r)}, \boldsymbol{b}^{(r)}, \boldsymbol{\Sigma}_{\mathtt{b}}^{(r)}$

Figure 6: Model-based `Joint` uncertainty decoding.

During decoding, in contrast to the front-end `Joint` scheme, all the front-end components representing different groups of recognition components are active and pass their measure of uncertainty to the recogniser. This operation is similar to using a multiple-transform constrained MLLR scheme [12], but with the addition of a variance bias.

3.3 Computational cost

The additional costs for different noise compensation schemes in the front-end processing, and during decoding are summarised in Table 1. The model-based uncertainty decoding scheme can be surprisingly efficient in comparison to the front-end, especially with diagonal variances. With the same number of transforms, $R = N$, the R parallel application of linear transforms to the features is more efficient than N Gaussian evaluations to determine the best component, although applying a diagonal linear transform and a Gaussian evaluation are both $\mathcal{O}(D)$. The main difference in performance however, is that the variance bias applied to the recognition model-set is fixed given a particular acoustic environment, in contrast to the front-end scheme where it will vary if either the acoustic environment or the front-end component changes, which is $\mathcal{O}(DTM)$.

Compensation Scheme	Front-end Cost	Decoding Cost
Feature Enhancement, e.g. `SPLICE`	$\mathcal{O}(DTN)$	None
Front-end Uncertainty Schemes	$\mathcal{O}(DTN)$	$\mathcal{O}(DTM)$
Model-based Uncertainty Schemes	$\mathcal{O}(DTR)$	$\mathcal{O}(DM)$
Model-based Forms, e.g. VTS	None	$\mathcal{O}(D^2M)$

Table 1: Summarising computational cost for different noise compensation schemes. D is number of feature dimensions, T the number of frames, N the number of front-end GMM components, R the number of acoustic model classes, and M the number of acoustic model components.

It is usually the case that the number of front-end model components N, or regression classes R, is far less than the number of Gaussians M in the actual recognition acoustic model. Thus any operation on the recognition model dominates the computational load. For uncertain observations and uncertainty decoding this is simply the addition of the diagonal variance bias, $\boldsymbol{\Sigma}_{\mathtt{b}}^{(n)}$, to each of the component variances, $\boldsymbol{\Sigma}^{(m)}$, and re-calculation of the determinant, a cost of $\mathcal{O}(D)$ per multivariate Gaussian. Unfortunately for front-end schemes, this bias will vary every time the front-end component changes, rather than when the acoustic environment changes. Though the parameters may all be cached, this starts to become impractical when for example a 256-component front-end model is used with a state-of-the-art recognition system with over 100,000 components. Therefore if model parameter updates can be retained[2], the model-based uncertainty decoding

[2] In some systems multiple audio channels are recognised against a single shared acoustic model loaded in memory, hence the model parameters cannot be updated if channels may host different acoustic environments.

form avoids the expensive $\mathcal{O}(DTM)$ variance updates in the acoustic model. The $\mathcal{O}(DTR)$ front-end transforms can also be saved if the model means may be cached as well.

Comparing this uncertainty decoding cost to, for example, the simplest form of model compensation, the Log-Add approximation, shows that the uncertainty decoding has the potential for large reductions in computational cost, depending on the number of times the front-end component changes. For the log-add approximations the dominant cost is a D-dimensional matrix vector multiplication for each recognition component, a cost of $\mathcal{O}(D^2)$. If the variances are compensated as well, the cost increases dramatically [13]. Using a truncated first-order VTS compensation scheme [1] requires the computation of two $D \times D$ matrices per Gaussian in the acoustic model and then several matrix multiplies to compensate. The model-based uncertainty form shares parameters for similar components, so that there is a great saving in their estimation; the compensation is also cheaper at $\mathcal{O}(DM)$ compared to $\mathcal{O}(D^2M)$.

4 Issues with uncertainty decoding

The previous chapter discussed one serious drawback with front-end uncertainty based schemes: that the model variances must be updated every time the variance bias changes. Although, the computation is simple compared to a technique such as model-based VTS compensation, it still involves an expensive re-computation of the typically cached Gaussian normalisation term. However, there is an even larger concern for front-end uncertainty decoding forms.

4.1 A fundamental problem

Consider the joint distribution of the clean speech and noise shown previously in figure 5. Two corrupted speech conditional distributions, $p(y|x)$, are marked. The first results when the SNR is relatively high, with the clean speech $x = 6$. This yields a highly skewed distribution that peaks sharply at $x = 6$ that is highly non-Gaussian, yet it is modeled with a normal distribution. This was shown to be not problematic in [19]. As the SNR increases this becomes more pronounced until it becomes a delta function yielding the clean speech distribution when substituted in equation 20. This is expected, since when the SNR is high, the noise should have no influence on compensating the acoustic models.

The corrupted speech conditional distribution looks very different when the SNR is low with $x = 1$ while $n = 3$. At this point, the distribution is Gaussian, matching the corrupting noise distribution, with a mean of 3 and variance of 1. Thus in low SNR, the conditional distribution approaches the noise distribution

$$p(\boldsymbol{y}_t|\boldsymbol{x}_t, \check{\mathcal{M}}) \approx \mathcal{N}(\boldsymbol{y}_t; \boldsymbol{\mu}_{\mathtt{n}}, \boldsymbol{\Sigma}_{\mathtt{n}}) \tag{43}$$

where $\boldsymbol{\mu}_{\mathtt{n}}$ and $\boldsymbol{\Sigma}_{\mathtt{n}}$ are the noise mean and variance respectively. This intuitively makes sense, since the noise masks the speech. This singularity has also been documented in [4] however the consequences for uncertainty decoding have not been previously examined. If equation 43 is substituted into equation 20, the distribution of the corrupted observation is the same as the noise distribution

$$p(\boldsymbol{y}_t|\mathcal{M}, \check{\mathcal{M}}, \theta_t) \approx \int \mathcal{N}(\boldsymbol{y}_t; \boldsymbol{\mu}_{\mathtt{n}}, \boldsymbol{\Sigma}_{\mathtt{n}}) p(\boldsymbol{x}_t|\mathcal{M}, \theta_t) d\boldsymbol{x}_t \tag{44}$$

$$= \mathcal{N}(\boldsymbol{y}_t; \boldsymbol{\mu}_{\mathtt{n}}, \boldsymbol{\Sigma}_{\mathtt{n}}) \tag{45}$$

since the conditional distribution is no longer a function of the clean speech.

Thus regardless of the original recognition model component, the compensated distribution used during decoding will always be identical to the noise distribution. When a single conditional distribution is estimated and used for all components, in low SNR conditions a frame, or sequence of frames, will have no discriminatory power between classes: Every distribution will look the same. If the recognition task has additional constraints beyond the acoustic models, such as a language model, then it may be possible to distinguish between different models during these non-discriminatory regions if these are applied. However, when there is no language model or other restrictions for example with a continuous digit recognition task such as AURORA, then these areas where no discriminatory acoustic information is available will be very susceptible to errors. These errors will probably be insertions since these are likely to be background regions, although low-energy speech may be substituted by other models if the noise is significant enough to mask the speech.

A clear illustration of this issue with the front-end `Joint` uncertainty decoding algorithm [19, 20] is presented in figure 7. This figure shows the clean speech, corrupted speech, front-end `Joint` estimate, given by $\boldsymbol{A}^{(n)}\boldsymbol{y}_t + \boldsymbol{b}^{(n)}$, and the standard deviation of the uncertainty variance bias, obtained from $\boldsymbol{\Sigma}_{\mathtt{b}}^{(n)}$, for a simple system with a 16-component front-end GMM. For those regions of higher energy speech, for example frames 180 to 190 where the vowel 'i' is articulated, the variance bias is small. On the other hand, in the lower energy regions around this vowel, for example frames 225 to 230, the variance becomes too large to be measured on this scale, as is the

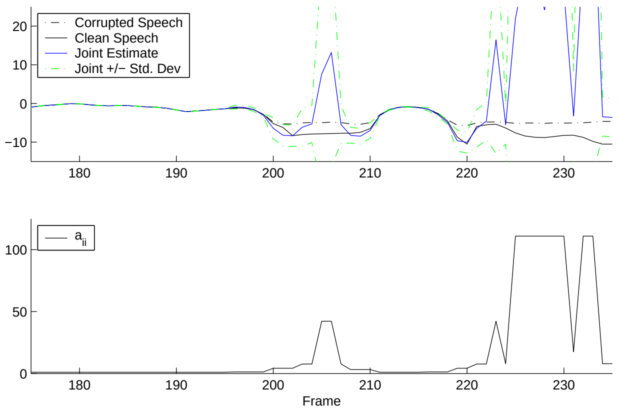

Figure 7: Plot of log energy for snippet from AURORA digit string 8-6-Zero-1-1-6-2, showing joint estimate and variance bias, along with the value of a_{ii}.

`Joint` estimate of the value. These large variances are associated with large values of the scale factor a_{ii} as shown in figure 7. In this example, from frames 225 to 230 the value of a_{ii} is around 100. With greater numbers of front-end components, these effects are amplified as parameters are no longer smoothed.

The reason that the magnitude of the scaling factor a_{ii}, and hence the variance biases, both become very large, can be ascertained by examining the nature of the joint distribution, as given in equation 29, in low energy speech regions. For regions with low SNR, the corrupted speech distribution is dominated by the noise; in other words, the noise masks the speech. Consider the cross-variance term $\mathbf{\Sigma}_{xy}^{(n)}$ for a front-end component associated with these regions of low speech energy

$$\mathbf{\Sigma}_{xy}^{(n)} = \mathbf{\Sigma}_{yx}^{(n)\mathsf{T}} = \mathcal{E}\left\{(\boldsymbol{x}_t - \boldsymbol{\mu}_x^{(n)})(\boldsymbol{y}_t - \boldsymbol{\mu}_y^{(n)})^{\mathsf{T}}\right\} = \mathbf{0} \tag{46}$$

The clean speech and the corrupted speech are uncorrelated since the clean speech and noise processes are independent. This lack of correlation drives $\boldsymbol{A}^{(n)}$, from equation 33, to infinity along with the model variance offsets. In front-end uncertainty decoding, this is expected behaviour because the front-end has determined that in these areas, the uncertainty is high, since the SNR is low. Given equation 46, the relationship to equation 45 becomes clearer by re-expressing equation 22, for component m, as

$$\begin{aligned} p\left(\boldsymbol{y}_t|\mathcal{M},\check{\mathcal{M}},\theta_t,m\right) &= \mathcal{N}\Big(\boldsymbol{y}_t;\mathbf{\Sigma}_{yx}^{(n)}\mathbf{\Sigma}_x^{(n)\text{-}1}\Big(\boldsymbol{\mu}^{(m)} - \boldsymbol{\mu}_x^{(n)}\Big) + \boldsymbol{\mu}_y^{(n)}, \\ &\qquad \mathbf{\Sigma}_{yx}^{(n)}\mathbf{\Sigma}_x^{(n)\text{-}1}\Big(\mathbf{\Sigma}^{(m)} + \mathbf{\Sigma}_x^{(n)}\Big)\mathbf{\Sigma}_x^{(n)\text{-}1}\mathbf{\Sigma}_{yx}^{(n)\mathsf{T}} + \mathbf{\Sigma}_y^{(n)}\Big) \qquad (47) \\ &= \mathcal{N}\left(\boldsymbol{y}_t;\boldsymbol{\mu}_y^{(n)},\mathbf{\Sigma}_y^{(n)}\right) \qquad (48) \end{aligned}$$

which is simply the noise distribution in a low energy region. Therefore, allowing an unconstrained estimate of $\boldsymbol{A}^{(n)}$ may result in large numbers of errors, mainly insertions, depending on the task.

Though this is correct in the sense that given the assumptions, this provides the compensation parameters to use, the assumptions are only simple approximations chosen to make the uncertainty decoding scheme efficient. Consequently, it may be prudent to mitigate the extreme symptoms that result by restraining the possible values for the compensation parameters. The obvious approach is to examine the correlation coefficients discussed earlier for each of the dimensions, defined as

$$\rho_{xy,i}^{(n)} = \frac{\sigma_{xy,i}^{(n)}}{\sqrt{\sigma_{x,i}^{(n)2}\sigma_{y,i}^{(n)2}}} \tag{49}$$

The compensation parameter estimates given in equations 33 to 35 can then be re-expressed in terms of the correlation coefficient as

$$a_{ii}^{(n)} = \frac{\sigma_{x,i}^{(n)}}{\rho_{xy,i}^{(n)}\sigma_{y,i}^{(n)}} \tag{50}$$

$$b_{i}^{(n)} = \mu_{x,i}^{(n)} - \frac{\sigma_{x,i}^{(n)}}{\rho_{xy,i}^{(n)}\sigma_{y,i}^{(n)}}\mu_{y,i}^{(n)} \tag{51}$$

$$\sigma_{\mathtt{b},i}^{(n)2} = \frac{\sigma_{x,i}^{(n)2}}{\rho_{xy,i}^{(n)2}} - \sigma_{x,i}^{(n)2} \tag{52}$$

for the diagonal form of front-end `Joint` uncertainty decoding. To restrict the extreme values of $a_{ii}^{(n)}$ and $\sigma_{\mathtt{b},i}^{(n)2}$ that can be obtained, a minimum value on the correlation coefficient can be enforced. Accordingly, the correlation $\rho_{xy,i}^{(n)}$ in equations 50-52 is set to

$$\hat{\rho}_{xy,i}^{(n)} = \max(\rho_{xy,i}^{(n)}, \rho) \tag{53}$$

where ρ is an empirically determined constant. Increasing the value of ρ raises the minimum acceptable correlation, decreasing the maximum variance bias. This can be viewed as enforcing a SNR floor; although the system may operate below this, the compensation scheme acts as though the floor is the actual level. In the limit, it is possible to set $\rho = 1$, which can be interpreted as assuming there is no noise in the environment, resulting in a zero variance bias in equation 52. SNR is highly related to correlation [5]—the correlation between speech and corrupted speech may be written as

$$\rho_{xy} = \frac{S}{\sqrt{S(S+N)}} \tag{54}$$

where S is the clean speech energy, and N the noise energy. The relationship between the correlation and SNR is clear when this is re-written as a ratio of the speech to noise

$$\frac{S}{N} = \frac{\rho_{xy}^2}{1-\rho_{xy}^2} \tag{55}$$

Equation 55 shows that a correlation of one equates to infinite SNR, and zero correlation corresponds to an SNR of negative infinity.

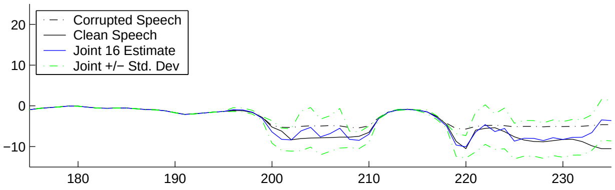

Figure 8: Plot of log energy for AURORA digit string 8-6-Zero-1-1-6-2, with correlation flooring, $\rho = 0.1$.

The effects of this flooring on the same snippet of artificially corrupted speech from figure 7 is shown in figure 8. As anticipated, the extremes in the variance bias observed before have been tempered.

This fundamental issue of all distributions becoming the same in low SNR theoretically affects all front-end uncertainty forms, the `SPLICE` with uncertainty decoding scheme should also suffer from it. However this issue has not been observed, for example, on the AURORA results presented in [9] with SPLICE. This is because SPLICE with uncertainty applies a limit on the maximum value of the variance bias scaling factor $a_{ii}^{(n)}$ to $1/\alpha$ in equation 28. Here α is also an empirically determined parameter. In addition to this explicit flooring, there is also an under-estimate of the value of $a_{ii}^{(n)}$. In order to make the calculation of the SPLICE uncertainty efficient, a global variance is used in the denominator of equation 25. Since this will be larger than any of the individual front-end components that should be used, the scaling estimate will be lower than expected as can be discerned from this equation. This under-estimation will become larger as the number of front-end components increases, since the variance of the individual model components will become smaller and smaller compared to the global variance. This is exactly the situation when a component might expected to be associated only with a low-energy noise region.

4.2 Comparison with other uncertainty-based schemes

In contrast, model-based compensation schemes do not suffer from this problem since all the models are not globally affected in the same manner. Typically each model, or group of components, is compensated individually and hence the relative affect of the corrupting noise is taken into account, for example that high-energy speech is less influenced by noise, then the original background models. This is a result of maintaining the tie between the conditional distribution and the clean speech distribution in equation 20. With each recognition component being compensated differently relative to the noise, each will be distinguishable from others, until the interfering noise subsumes all possible speech. Therefore, this theoretical issue with all front-end uncertainty based techniques is not present for model-based forms.

With observation uncertainty, this is also not an issue if appropriate models are used. If a naïve form of the clean speech posterior is used, then the uncertainty in overwhelming noise with be infinite and the same problems with arise. However, with SPLICE and model-based feature enhancement, models of the clean speech are used in the front-end processing. In the latter, the MMSE estimate in 18 and variance in equation 19 applied to the acoustic models, where the cross correlation again is zero, becomes

$$\hat{\boldsymbol{x}}_t^{(n)} = \boldsymbol{\mu}_x^{(n)} \tag{56}$$

$$\boldsymbol{\Sigma}_{\hat{\mathbf{x}}}^{(n)} = \boldsymbol{\Sigma}_x^{(n)} - \boldsymbol{\Sigma}_{xy}^{(n)}\left(\boldsymbol{\Sigma}_y^{(n)}\right)^{-1}\boldsymbol{\Sigma}_{yx}^{(n)} \tag{57}$$

$$= \boldsymbol{\Sigma}_x^{(n)} \tag{58}$$

which in this case is the simplified clean speech model for this noise region n, which in low SNR *is* the noise variance. Obviously, substituting this clean speech posterior into equation 16 is not problematic. Similarly, the variance of the Weiner filtering process yields the noise variance when the SNR approaches $-\infty$ [4]. Despite assertions that both observation uncertainty and uncertainty decoding are valid interpretations [2], the feature uncertainty converging to the noise variance in low SNR does not make sense. Intuition suggests that if the noise subsumes speech, then there should be no certainty in the feature vector, and therefore the observation uncertainty should be infinite. This inconsistency demonstrates a flaw in the observation uncertainty approach.

4.3 Summary

In this section, a major issue for front-end uncertainty decoding forms has been discussed. There can be low SNR regions where there is no acoustic differentiation possible which can result in spurious insertion errors if no other search constraints are available. This was demonstrated on two such front-end uncertainty forms: SPLICE and Joint. A possible remedy for the Joint form was proposed, and why SPLICE with uncertainty does not explicitly display these problems discussed. In comparison to these forms, the observation uncertainty approach does not have this inherent issue, but itself is not mathematically sound. Also pure model-based techniques like PMC or ALGONQUIN do not suffer from this problem, as they compensation each component differently taking into account the non-linear affect of the noise depending on the level of the speech.

5 Experiments

This section reports quantitative results on the standard small-vocabulary AURORA task and the Resource Management corpus—both artificially corrupted databases.

5.1 The AURORA system

AURORA 2.0 is a small vocabulary digit string recognition task [14]. Utterances are one to seven digits long based on the TIDIGITS database with noise artificially added. The clean training data is comprised of 8440 utterances with 55 male and 55 female speakers. For matched training, 422 sentences are provided for each of 16 conditions: 4 different SNRs ranging from 20 to 5 dB, and with the 4 different additive noise sources N1 to N4: subway, babble, car and exhibition hall. Each of the 16 conditions also has a test set of a 1001 sentences with 52 male and 52 female speakers.

The reference recogniser uses a 39 dimensional feature vector consisting of 12 MFCCs appended with unnormalised log energy, delta and delta-delta coefficients. The acoustic models are whole word digit models, each with 16 emitting states, 3 mixtures per state and silence and inter-word pause models. For this work, the HTK 3.3 internal alpha system was used, as opposed to the reference 2.2 version, along with its native front-end processing code [26]. This resulted in very minor differences in the baseline performance.

5.2 The Resource Management system

The 1000 word naval ARPA Resource Management (RM) database [22] was corrupted with noise at the waveform level from the NOISEX-92 database. The clean data was recorded in a sound-isolated room using a head mounted Sennheiser HMD414 noise-canceling microphone yielding a high signal-to-noise ratio of 49 dB[3]. The speech was recorded with 16 bit resolution at 20 kHz and down-sampled subsequently to 16kHz. The speaker independent training data for this task consists of 109 speakers reading 3990 sentences of prompted script. The utterances vary in length from about 3 to 5 seconds totaling 3.8 hours of data.

The NATO NOISEX-92 database provides recording samples of various artificial, pedestrian and military noise environments recorded at 20 kHz with 16 bit resolution. The Destroyer Operations Room noise was sampled at random intervals and added to the clean speech data at the waveform level prior to parameterisation. Figure 9 shows the affect of the noise on one of the RM sentences "Clear all windows". The noise itself has a dominant low frequency background hum, an unknown repetitive 6 Hz broadband noise of a machine, and intermittent speech.

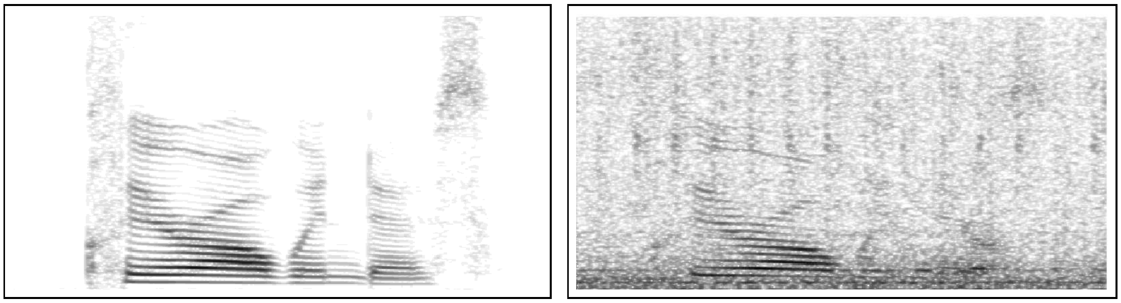

Figure 9: Clean spectrum (left) compared to with Operating Room noise at 8 dB SNR (right) "Clear all windows".

The baseline recogniser was built using the RM recipe distributed with HTK [26]. The 39 dimensional feature vector consists of 12 MFCCs appended with the log energy, velocity and

[3] The `wavmd` tool from the NIST Speech Quality Assurance Package v2.3 was used to determine the SNR.

acceleration coefficients. The cross-word, state-clustered triphone acoustic models with six components per state were used along with a simple bigram word pair grammar. All results are quoted as an average of three of the four available test sets, `Feb'89`, `Oct'89` and `Feb'91`, unless otherwise stated; the `Sep'92` test data was not used. This gave a total of 30 test speakers and 900 utterances. All decoding experiments were run using this system as the standard RM configuration unless otherwise stated.

5.3 Front-end model estimation

An important issue is the generation of the front-end GMM and the process used to obtain the compensation parameters. Two sets of GMMs for the front-end uncertainty models were trained using iterative mixture splitting. The first set used the clean speech data to train the models. For uncertainty decoding schemes described here, this is the preferred way of building models, since provided a noise model is available or can be estimated, the compensation parameters can be simply estimated using VTS or PMC style approaches (for more details see [19]). This is similar to the method taken in model-based feature enhancement [25] where the front-end corrupted speech GMM is derived from a clean speech GMM given a single Gaussian model of the additive noise using VTS compensation. In [24] it was found that a more complex phoneme based front-end HMM actually degraded performance as errors were detrimentally propagated to the decoding process. The second set of models were directly estimated from corrupted speech data. This should better represent the corrupted speech acoustic space.

For the model-based schemes, the GMM was not trained explicitly, but rather each Gaussian is linked with recognition model parameters. When using these model-based schemes the parameters in, for example, equation 40 were obtained from the clean data. The compensation parameters for all schemes were estimated using stereo data for the specific noise condition. This allows the techniques to be assessed without having to consider inaccuracies that result from the noise estimation process, or approximations in the mismatch function. In practical situations where stereo data is not available, the compensation parameters can be estimated using PMC or VTS style schemes [19]. For the front-end uncertainty schemes only diagonal transformations were used.

5.4 Results

Table 2 shows the baseline word error rates along with `SPLICE` system performance on the AURORA task. As usual, the addition of noise seriously degrades the performance of the system unless the clean models are compensated. The matched, approximate "best", target performance is also shown; this matched system was built using stereo data and single-pass re-training [11] to maintain the clean speech transition probabilities, but update the output distributions to reflect the corrupted speech. `SPLICE` was evaluated with both clean and corrupted speech GMMs. The results presented in the table are with a 256-component GMMs, but the same general trends were observed for both more and less components. Not surprisingly the use of the corrupted speech trained GMM, as presented in [9], outperformed the clean speech trained GMM. It is curious that the `SPLICE` with uncertainty schemes were so sensitive to the choice of GMM. However this may be an attribute of the limitations of the front-end schemes discussed in section 4. To investigate the effects of the flooring on `SPLICE` with uncertainty a range of values of α (see equation 28) were tried. Table 2 shows the performance for 0.1 (as recommended in [9]) and the best observed over the range of SNRs 0.95. By increasing the value of α from 0.1 to 0.95 slight performance gains were obtained, especially on the lower SNR conditions. The best configuration was `SPLICE` with uncertainty and $\alpha = 0.95$.

Table 3 shows the performance of the front-end `Joint` scheme. Two configurations were run. The first used no flooring value for ρ. In contrast to the RM system in [19] where significant performance gains were obtained with no ρ flooring, the performance was significantly worse than the baseline `SPLICE` system. While slightly fewer deletions and substitutions occurred overall, a vast number of insertions appeared in regions where there were a series of frames with low-correlation co-efficients. For example, on the car noise at 20 dB, using the `Joint` uncertainty

System	SNR(dB)			
	20	15	10	5
Clean	4.62	12.20	31.13	59.16
Matched	1.85	2.81	5.01	11.41
Clean Speech GMM				
`SPLICE`	1.97	2.96	6.24	15.74
+Uncertainty, $\alpha = 0.1$	2.49	4.13	8.88	23.06
+Uncertainty, $\alpha = 0.95$	2.30	3.88	8.30	21.38
Corrupted Speech GMM				
`SPLICE`	1.95	3.07	6.13	16.47
+Uncertainty, $\alpha = 0.1$	2.15	3.22	5.95	14.50
+Uncertainty, $\alpha = 0.95$	2.00	3.20	5.58	12.29

Table 2: Clean, matched and `SPLICE`with 256 components systems' performance on AURORA 2.0 test set A, averaged across N1-N4, WER(%).

decoding form with 256 components and the clean speech GMM, the number insertion errors was 421, a magnitude increase from 31 when no noise is present. If ρ is set to 0.9, this drops to a reasonable 19 insertion errors, compared to a total of 13 for the matched system. The correlation floor at this level gives significantly improved performance for both the clean and corrupted speech trained GMM systems. The final front-end `Joint` scheme is comparable to the best `SPLICE` with uncertainty approach for both clean and corrupted GMM systems.

System	SNR(dB)			
	20	15	10	5
Clean Speech GMM				
FE-`Joint`	16.99	20.50	25.95	42.78
FE-`Joint`, $\rho = 0.9$	1.93	2.98	6.09	16.36
Corrupted Speech GMM				
FE-`Joint`	22.67	25.82	28.38	34.37
FE-`Joint`, $\rho = 0.9$	1.81	2.88	5.71	14.62

Table 3: 256-component front-end `Joint`systems' performance on AURORA 2.0 test set A, averaged across N1-N4, WER(%).

To present a more detailed view of how individual frames and elements in each of those frames are affected by this flooring, results of a 16-component simplified system are presented in Table 4. When ρ is in greater than 0.9 all the low-energy and background related coefficients are affected, severely restraining the magnitudes of the mean and variance biases. Nevertheless, this appears to be an effective strategy.

Strategy	ρ floor						
	0.99	0.95	0.9	0.5	0.1	0.01	-1.0
% Frames affected	100	100	100	58	33	27	0
% Elements affected	99	97	90	46	28	12	0
WER	2.79	2.52	2.52	3.75	24.47	20.82	20.17

Table 4: Flooring $\rho_{xy,i}$ on 16-component FE-`Joint`System on AURORA 20dB SNR, WER(%)

The model-based `Joint` approach was examined on this task with results reported in Table 5. Five systems were built. The first three used diagonal transformations, similar to the front-end

System	Number of Transforms	SNR(dB)			
		20	15	10	5
Diagonal Transformations					
M-Joint	1	3.33	5.92	13.35	31.96
	16	2.47	3.82	7.25	16.63
	256	1.90	2.73	5.19	12.00
Full Transformations					
M-Joint	1	2.43	3.82	6.97	17.14
	16	1.95	2.80	4.23	9.89

Table 5: Model-based Jointsystems' performance on AURORA 2.0 test set A, averaged across N1-N4, WER(%).

schemes. The performance of the 16 transformation model-based Joint scheme was slightly worse than the appropriately floored 256 component front-end schemes, but at considerably reduced computational cost; with the same number of diagonal transforms, 256, the model-based system is far superior to the front-end version. Moreover, using full transformations gave considerable gains. The 16-transform full variance model-based approach yielded better performance at low SNR than the matched system. However as the variance bias is a full matrix, there is the high cost of performing a full covariance matrix decode, compared to the diagonal covariance matched system. This does indicate an opportunity to obtain excellent performance using this model-based Joint approach.

On the RM task, as shown in Table 6, the incorporation of this correlation flooring does not affect performance, until it is severely set to 0.9, where on this task, at this level, it degrades performance. The presence of a language model to guide the recognition during low SNR regions makes the flooring unnecessary. The corrupted speech GMM in this FE-Joint system was derived from a clean speech GMM and single-pass re-training rather than directly from the corrupted speech data.

System	# of Comps.	ρ_{const} Floor				
		0.9	0.5	0.1	0.01	-1.0
Clean	—	33.2				
FE-Joint	16	10.8	9.3	9.8	9.7	9.8
	256	10.3	8.2	8.2	8.4	8.4
Matched	—	7.2				

Table 6: Flooring $\rho_{xy,ii}$ on FE-JointSystem on RM 20dB SNR, WER(%)

Lastly, results of model-based Joint uncertainty decoding on RM are presented in Table 7. As in the AURORA results, greater transform specificity improves results; however, increasing the number of transforms beyond 16 did not affect performance much. Also, we similarly find that the most powerful full Joint transform systems exceed matched system performance. This essentially incorporates the correlations between dimensions while using diagonal acoustic model variances.

	# of	Transform Kind	
System	Trans.	Diagonal	Full
Clean	—	33.2	
Model-Joint	16	8.2	7.4
	256	8.0	7.4
Matched	—	7.2	

Table 7: Model-JointSystem on RM 20dB SNR, WER(%)

6 Conclusions

This report has presented a fundamental problem with front-end uncertainty decoding methods: by only propagating a single vector of features and probabilities, during periods where the noise is predominant, the ability to effectively discriminate can be lost. When all the models become identical in these situations, this causes insertion errors in the search. With another source for discrimination, such as a language model, this can be less of an issue as it guides the search when the SNR is low and the uncertainty is high. For the Joint compensation scheme, a correlation floor can be used to enforce a bound on the uncertainty decoding scaling, ensuring that all models are not updated to be the same. In the SPLICE with uncertainty formulation, the flooring of the variance of the clean speech posterior and the use of a global clean speech prior, both aid in preventing this issue from occurring. Model-based uncertainty decoding approaches do not suffer from this problem since the corrupted speech conditional is tied to the clean speech acoustic model. This ensures each recognition component, or group of components, is compensated differently depending much the noise affects them. If the uncertainty parameters can be shared across classes of similar recognition components, such as with the model-based Joint uncertainty form, efficiency is similar to the front-end versions, without this fundamental problem.

These factors were explored on the small vocabulary AURORA and thousand-word RM databases. The need to floor the correlations was demonstrated for front-end uncertainty decoding such as SPLICE and Joint forms on the AURORA task; these two algorithms perform comparably. On this small task, the model-based Joint uncertainty form was superior to either. Similar trends were observed on the medium vocabulary RM database, however the correlation flooring was not necessary due to the presence of a language model to guide the search in low SNR areas. Overall, the Joint model-based form proved to be a superior uncertainty decoding form, achieving better noise robustness with a pleasing computational profile compared to the other forms evaluated.

A major limitation of this paper is that experiments are all conducted on artificially corrupted data and assume stationarity of the noise. This limitation is addressed in [21] where a process for estimating noise models for Joint uncertainty decoding is presented along with preliminary results using found data such as Broadcast News.

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] J. A. Arrowood. *Using Observation Uncertainty for Robust Speech Recognition.* PhD thesis, Georgia Institute of Technology, 2003.

[3] J. A. Arrowood and M. A. Clements. Using Observation Uncertainty In HMM Decoding. In *Proc. ICSLP*, Denver, Colorado, Sept. 2002.

[4] 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.

[5] M. Borga. Canonical correlation, a tutorial, Jan. 2001. Available from: http://people.imt.liu.se/˜magnus/cca/.

[6] 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.

[7] L. Deng, J. Droppo, and A. Acero. Exploiting variances in robust feature extraction based on a parametric model of speech distortion. In *Proc. ICSLP*, 2002.

[8] L. Deng, J. Droppo, and A. Acero. Dynamic compensation of HMM variances using the feature enhancement uncertainty computed from a parametric model of speech distortion. *IEEE Transactions on Speech and Audio Processing*, 12(3), May 2005.

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

[10] 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.

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

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

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

[14] 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.

[15] J. N. Holmes, W. J. Holmes, and P. N. Garner. Using formant frequencies in speech recognition. In *Proc. Eurospeech*, Rhodes, Greece, Sept. 1997.

[16] 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.

[17] 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.

[18] T. T. Kristjansson and B. J. Frey. Accounting for uncertainty in observations: A new paradigm for robust speech recognition. In *Proc. ICASSP*, Orlando, Florida, May 2002.

[19] 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.

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

[21] H. Liao and M. J. F. Gales. Joint uncertainty decoding for robust large vocabulary speech recognition. Technical Report CUED/F-INFENG/TR552, University of Cambridge, 2006. Available from: mi.eng.cam.ac.uk/˜hl251.

[22] 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*, Seattle, Washington, USA, May 1988.

[23] K. Shinoda and T. Watanabe. Speaker adaptation with autonomous control using tree structure. In *Proc. Eurospeech*, pages 1143–1146, Madrid, Spain, Sept. 1995.

[24] 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.

[25] V. Stouten, H. V. hamme, and P. Wambacq. Accounting for the uncertainty of speech estimates in the context of model-based feature enhancement. In *Proc. ICSLP*, volume I, pages 105–108, Jeju Island, Korea, Oct. 2004.

[26] 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.