Model-based approaches to adaptive training in reverberant environments

Y.-Q. Wang and M.J.F. Gales

Engineering Department, Cambridge University
Trumpington St. Cambridge University, CB2 1PZ, U.K.
{yw293,mjfg}@eng.cam.ac.uk

Abstract

Adaptive training is a powerful approach for building speech recognition systems using non-homogeneous data. This work presents an extension of model-based adaptive training to handle reverberant environments. The recently proposed Reverberant VTS-Joint (RVTSJ) adaptation[1] is used to factor out unwanted additive and reverberant noise variations in multi-conditional training data, yielding a canonical model neutral to noise conditions. An maximum likelihood estimation of the canonical model parameters is described. An initialisation scheme that uses the VTS-based adaptive training to initialise the model parameters is also presented. Experiments are conducted on a reverberant simulated AURORA4 task.

Index Terms: reverberant noise robustness, vector Taylor series, adaptive training

1. Introduction

Model-based approaches for noise-robust speech recognition have been investigated and extended in a number of ways, e.g. Vector Taylor series (VTS) compensation[2, 3] and joint uncertain decoding (JUD)[2]. However, there has been less work in applying model-based approaches to handle noise in reverberant environments. In [1, 4], two model compensation schemes, reverberant VTS (RVTS) and reverberant VTS-joint (RVTSJ), are proposed, where the underlying acoustic model is adapted to the target environment given an estimated noise model. The RVTSJ compensation enables the joint estimation of additive and reverberant noise, thus gives better performance. RVTS and RVTSJ schemes assume the underlying acoustic model is trained on clean data. Acoustic models trained on multi-conditional data generally give better noise robustness. In [5], a clean corpus was filtered by several room impulse responses (RIRs) to form the multi-conditional data. It was found that the multi-style trained acoustic model (MST) followed by general adaptation, e.g., MLLR, yielded large gains. Recently, [6] shows that using stereo data in multi-style training can further improve the reverberant noise robustness. However, multi-style training forces the noise variations to be modelled by the underlying acoustic model, which could potentially harm the performance. It also degrades the performance when the acoustic model is operated outside the training environments.

Alternatively, adaptive training can be applied to factor out the unwanted noise variation, yielding a canonical model that is neutral to the noise condition. Adaptive training is originally proposed for speaker adaptive training[7], and is extended to handle additive and convolutional noise recently[2, 8]. Motivated by the success of VTS-based adaptive training (VAT) [8], this work investigates the model-based approaches to adaptive training in reverberant environments. RVTSJ is used to compensate acoustic models in both training and testing. The new training algorithm is referred to reverberant adaptive training (RAT). An maximum likelihood (ML) estimation of canonical model parameters in the EM framework is described. An initialisation scheme using VAT canonical model parameters to start RAT training is also presented. This RAT training algorithm was compared with MST and VAT on a reverberant simulated version of AURORA4.

This work was partially supported by Google research award and DARPA under the GALE and RATS program. The paper does not necessarily reflect the position or the policy of US Government and no official endorsement should be inferred.

2. Reverberant and additive noise

If clean speech is corrupted by additive and short-time channel distortion (a.k.a. convolutional noise) in the Mel-cepstral domain, the mismatch function describing the distortion on the current clean speech parameter $\boldsymbol{x}_t$ is given by :

$$\begin{aligned}\boldsymbol{y}_t^{\mathtt{s}} =& \mathbf{C}\log\left(\exp(\mathbf{C}^{-1}(\boldsymbol{x}_t^{\mathtt{s}}+\boldsymbol{\mu}_{\mathtt{h}}))+\exp(\mathbf{C}^{-1}\boldsymbol{n}_t^{\mathtt{s}})\right)\\ =& \mathbf{f}(\boldsymbol{x}_t^{\mathtt{s}},\boldsymbol{\mu}_{\mathtt{h}},\boldsymbol{n}_t^{\mathtt{s}}),\end{aligned} \tag{1}$$

where the superscript (and also the subscript hereafter) $\mathtt{s}$ denotes static parameter, and $\boldsymbol{y}_t$, $\boldsymbol{h}$ and $\boldsymbol{n}_t$ are the noisy speech, convolutional and additive noise, respectively, $\mathbf{C}$ is the (truncated) DCT matrix. It is usually assumed that $\boldsymbol{n}$ is Gaussian distributed with mean $\boldsymbol{\mu}_{\mathtt{n}}$ and diagonal matrix $\boldsymbol{\Sigma}_{\mathtt{n}}$, and $\boldsymbol{h}$ is a unknown constant. Note that the above mismatch function relies on the assumption that the effective length of the impulse response of the channel is shorter than the size of the analysis window (typically 25ms). However, in a reverberant environment, due to the late-reflection caused by multiple acoustic paths from the speaker to the microphone, the reverberant time T_{60} which is the time needed for reflections sound to decay 60dB, is usually ranging from 200ms to 800ms or even longer. This is significantly larger than the size of analysis window. The long reverberant time causes the clean speech not only distorted by the additive noise, but also blurred by several previous frames. In [1], a mismatch function describing the joint effect of additive and reverberant noise is derived:

$$\begin{aligned}\boldsymbol{z}_t^{\mathtt{s}} &= \mathbf{C}\log\left(\sum_{\delta=0}^{n}\exp\left(\mathbf{C}^{-1}(\boldsymbol{x}_{t-\delta}^{\mathtt{s}}+\boldsymbol{\mu}_{\mathtt{l}\delta})\right)+\exp\left(\mathbf{C}^{-1}\boldsymbol{n}_t^{\mathtt{s}}\right)\right)\\ &= \mathbf{g}(\boldsymbol{x}_t^{\mathtt{s}},\cdots,\boldsymbol{x}_{t-n}^{\mathtt{s}},\boldsymbol{\mu}_{\mathtt{l}},\boldsymbol{n}_t^{\mathtt{s}})\end{aligned} \tag{2}$$

where $\boldsymbol{z}_t$ is the additive and reverberant noise corrupted speech, $\boldsymbol{\mu}_{\mathtt{l}}=[\boldsymbol{\mu}_{\mathtt{l}0}^{\mathsf{T}},\cdots,\boldsymbol{\mu}_{\mathtt{l}n}^{\mathsf{T}}]^{\mathsf{T}}$ is referred to as reverberant noise.

2.1. Model compensation

Given the additive and convolutional noise mismatch function in Eq. (1), VTS can be used to approximate it for every Gaus-

sian component m in the following form:

$$\boldsymbol{y}_t^{\mathsf{s}}|m \approx \mathbf{f}(\boldsymbol{\mu}_{\mathtt{sx}}^{(m)}, \boldsymbol{\mu}_{\mathtt{h}}, \boldsymbol{\mu}_{\mathtt{sn}}) + \mathbf{J}_{\mathtt{x}}^{(m)}(\boldsymbol{x}_t^{\mathsf{s}} - \boldsymbol{\mu}_{\mathtt{sx}}^{(m)}) + \mathbf{J}_{\mathtt{n}}^{(m)}(\boldsymbol{n}_t^{\mathsf{s}} - \boldsymbol{\mu}_{\mathtt{n}}),$$

where $\boldsymbol{\mu}_{\mathtt{n}}$ and $\boldsymbol{\mu}_{\mathtt{x}}^{(m)}$ are the means of additive noise and m-th Gaussian respectively, and

$$\mathbf{J}_{\mathtt{x}}^{(m)} = \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\Big|_{\boldsymbol{\mu}_{\mathtt{sx}}^{(m)}, \boldsymbol{\mu}_{\mathtt{sn}}, \boldsymbol{\mu}_{\mathtt{h}}}, \quad \mathbf{J}_{\mathtt{n}}^{(m)} = \mathbf{I} - \mathbf{J}_{\mathtt{x}}^{(m)} \tag{3}$$

This yields the following VTS-based model compensation form for the static parameter:

$$\boldsymbol{\mu}_{\mathtt{sy}}^{(m)} = \mathbf{f}(\boldsymbol{\mu}_{\mathtt{sx}}^{(m)}, \boldsymbol{\mu}_{\mathtt{h}}, \boldsymbol{\mu}_{\mathtt{n}}), \boldsymbol{\Sigma}_{\mathtt{sy}}^{(m)} = \mathrm{diag}\left(\mathbf{J}_{\mathtt{x}}^{(m)} \boldsymbol{\Sigma}_{\mathtt{sx}}^{(m)\mathsf{T}} \mathbf{J}_{\mathtt{x}}^{(m)}\right) \tag{4}$$

The dynamic parameter compensation can be derived using the continuous time approximation, i.e.,

$$\boldsymbol{\mu}_{\Delta\mathtt{y}}^{(m)} = \mathbf{J}_{\mathtt{x}}^{(m)} \boldsymbol{\mu}_{\Delta\mathtt{x}}^{(m)}, \ \boldsymbol{\Sigma}_{\Delta\mathtt{y}}^{(m)} = \mathrm{diag}\left(\mathbf{J}_{\mathtt{x}}^{(m)} \boldsymbol{\Sigma}_{\Delta\mathtt{x}}^{(m)\mathsf{T}} \mathbf{J}_{\mathtt{x}}^{(m)}\right) \tag{5}$$

where the subscript Δ denotes the delta parameter. The delta-delta parameter is compensated in a similar way. For notation convenience, only the delta parameter is considered in the following discussion.

For the reverberant and additive noise mismatch function in Eq. (2), the current observation $\boldsymbol{z}_t$ is a function of the preceding clean speech frames $\boldsymbol{x}_{t-1}, \cdots, \boldsymbol{x}_{t-n}$ (ignoring the dynamic parameters). These clean speech frames should be inferred from the noise corrupted observations. In practice, this is computationally intractable. An approximation form was proposed in [4], where $\boldsymbol{z}_t$ is assumed to depend on an extended vector $\overline{\boldsymbol{x}}_t$, which is generated by the current Gaussian component. Figure 1 illustrates the dynamic Bayesian network of this approximated model, where q_t and w_t are the indicator of current state and Gaussian component respectively.

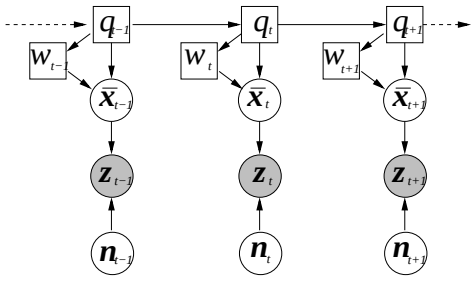

Figure 1: Approximate reverberant dynamic Bayesian network. For clarity, the dynamic parameters are ignored.

The form of $\overline{\boldsymbol{x}}_t$ is chosen such that when there is no reverberant noise, the compensated model will back off to the standard VTS compensation, and the Gaussian distribution is used to model $\overline{\boldsymbol{x}}_t$ conditioning on the current component m, i.e.,

$$\overline{\boldsymbol{x}}_t = \begin{bmatrix} \boldsymbol{x}_t \\ \Delta \boldsymbol{x}_t \\ \Delta^2 \boldsymbol{x}_t \\ \Delta^3 \boldsymbol{x}_t \\ \vdots \end{bmatrix} = \mathbf{W} \begin{bmatrix} \boldsymbol{x}_{t+w} \\ \cdots \\ \boldsymbol{x}_{t-n-w} \end{bmatrix}; \tag{6}$$

and $\overline{\boldsymbol{x}}_t|m \sim \mathcal{N}(\overline{\boldsymbol{\mu}}_{\mathtt{x}}^{(m)}, \overline{\boldsymbol{\Sigma}}_{\mathtt{x}}^{(m)})$, where w is the window size to calculate the dynamic parameter, $\mathbf{W}$ is a square and invertible matrix, which maps a sequence of statics to static plus the first, second and higher order dynamics. Given the model statistics $\overline{\boldsymbol{\mu}}_{\mathtt{x}}^{(m)}$, it is easy to derive the statistics of spliced vector $\boldsymbol{x}_{\mathtt{e}}^{\mathsf{s}} = (\boldsymbol{x}_t^{\mathsf{s}\mathsf{T}}, \cdots, \boldsymbol{x}_{t-n}^{\mathsf{s}\mathsf{T}})^{\mathsf{T}}$. For example, the static and delta mean vectors of $\boldsymbol{x}_{\mathtt{e}}$ can be obtained by

$$\boldsymbol{\mu}_{\mathtt{sxe}}^{(m)} = \mathcal{E}\{\boldsymbol{x}_{\mathtt{e}}^{\mathsf{s}}|m\} = \mathbf{P}_{\mathtt{s}} \overline{\boldsymbol{\mu}}_{\mathtt{x}}^{(m)}, \boldsymbol{\mu}_{\Delta\mathtt{xe}}^{(m)} = \mathcal{E}\{\Delta \boldsymbol{x}_{\mathtt{e}}|m\} = \mathbf{P}_{\Delta} \overline{\boldsymbol{\mu}}_{\mathtt{x}}^{(m)} \tag{7}$$

where $\mathbf{P}_{\mathtt{s}}$ and $\mathbf{P}_{\Delta}$ are the matrices that map $\overline{\boldsymbol{x}}_t$ to $\boldsymbol{x}_{\mathtt{e}}^{\mathsf{s}}$ and $\Delta \boldsymbol{x}_{\mathtt{e}}$.

Using statistics of the extended vector, VTS is extended in [1] to handle the mismatch function in Eq. (2). The expansion is performed at $(\boldsymbol{\mu}_{\mathtt{sxe}}^{(m)}, \boldsymbol{\mu}_{\mathtt{1}}, \boldsymbol{\mu}_{\mathtt{n}})$:

$$\boldsymbol{z}_t^{\mathsf{s}}|m \approx \mathbf{g}(\boldsymbol{\mu}_{\mathtt{sxe}}^{(m)}, \boldsymbol{\mu}_{\mathtt{1}}, \boldsymbol{\mu}_{\mathtt{n}}) + [\mathbf{J}_{\mathtt{xe}}^{(m)}, \mathbf{J}_{\mathtt{ne}}^{(m)}] \begin{bmatrix} \boldsymbol{x}_{\mathtt{e}}^{\mathsf{s}} - \boldsymbol{\mu}_{\mathtt{sxe}}^{(m)} \\ \boldsymbol{n}_t^{\mathsf{s}} - \boldsymbol{\mu}_{\mathtt{n}} \end{bmatrix} \tag{8}$$

where

$$\mathbf{J}_{\mathtt{xe}}^{(m)} = [\mathbf{J}_{\mathtt{x}0}^{(m)}, \cdots, \mathbf{J}_{\mathtt{x}n}^{(m)}]; \quad \mathbf{J}_{\mathtt{ne}}^{(m)} = \mathbf{I} - \sum_{\delta=0}^{n} \mathbf{J}_{\mathtt{x}\delta}^{(m)} \tag{9}$$

and $\mathbf{J}_{\mathtt{x}\delta}^{(m)} = \frac{\partial \mathbf{g}}{\partial \boldsymbol{x}_{t-\delta}^{\mathsf{s}}}\Big|_{\boldsymbol{\mu}_{\mathtt{sxe}}^{(m)}, \boldsymbol{\mu}_{\mathtt{1}}, \boldsymbol{\mu}_{\mathtt{n}}}$. Along with the continuous time approximation assumption, this yields the mean compensation form in the following:

$$\boldsymbol{\mu}_{\mathtt{sz}}^{(m)} = \mathbf{g}(\boldsymbol{\mu}_{\mathtt{sxe}}^{(m)}, \boldsymbol{\mu}_{\mathtt{1}}, \boldsymbol{\mu}_{\mathtt{n}}), \quad \boldsymbol{\mu}_{\Delta\mathtt{z}}^{(m)} = \sum_{\delta=0}^{n} \mathbf{J}_{\mathtt{x}\delta}^{(m)} \boldsymbol{\mu}_{\Delta\mathtt{x}\delta}^{(m)} \tag{10}$$

This is referred to as RVTSJ. It is possible to compensate the variance as well. However, in the initial investigation in [1], it was found that variance compensation is not quite effective, thus the standard VTS variance compensation was used, i.e., $\boldsymbol{\Sigma}_{\mathtt{z}}^{(m)} = \boldsymbol{\Sigma}_{\mathtt{y}}^{(m)}$. This is also adopted in this work.

2.2. Noise estimation

Given the acoustic model parameter $\mathcal{M} = \{\overline{\boldsymbol{\mu}}_{\mathtt{x}}^{(m)}, \boldsymbol{\Sigma}_{\mathtt{x}}^{(m)}\}$, the noise model parameter $\boldsymbol{\Phi} = (\boldsymbol{\mu}_{\mathtt{1}}, \boldsymbol{\mu}_{\mathtt{n}})$ is estimated using EM. The following auxiliary function is maximised, i.e.,

$$\hat{\boldsymbol{\Phi}} = \arg\max \sum_{t,m} \gamma_t^{(m)} \log p(\boldsymbol{z}_t; \boldsymbol{\mu}_{\mathtt{z}}^{(m)}, \boldsymbol{\Sigma}_{\mathtt{z}}^{(m)}) \tag{11}$$

where $\gamma_t^{(m)}$ is the posterior of component m at time t, given the current hypothesis and current noise estimates $\boldsymbol{\Phi}$. VTS is again used to expand $\boldsymbol{\mu}_{\mathtt{z}}^{(m)}$ using the current noise estimates $\boldsymbol{\Phi}$, yielding the following update formula using the second-order method:

$$\begin{aligned} \begin{bmatrix} \hat{\boldsymbol{\mu}}_{\mathtt{1}} \\ \hat{\boldsymbol{\mu}}_{\mathtt{n}} \end{bmatrix} &= \left(\sum_{t,m} \gamma_t^{(m)} \mathbf{J}^{(m)\mathsf{T}} \boldsymbol{\Sigma}_{\mathtt{sz}}^{(m)-1} \mathbf{J}^{(m)} + \alpha \mathbf{I} \right)^{-1} \\ &\times \left(\sum_{t,m} \gamma_t^{(m)} \mathbf{J}^{(m)\mathsf{T}} \boldsymbol{\Sigma}_{\mathtt{sz}}^{(m)-1} \left(\boldsymbol{\mu}_{\mathtt{sz}}^{(m)} - \mathbf{J}^{(m)} \begin{bmatrix} \boldsymbol{\mu}_{\mathtt{1}} \\ \boldsymbol{\mu}_{\mathtt{n}} \end{bmatrix} \right) \right) \end{aligned} \tag{12}$$

where

$$\mathbf{J}^{(m)} = [\mathbf{J}_{\mathtt{1}0}^{(m)}, \cdots, \mathbf{J}_{\mathtt{1}n}^{(m)}, \mathbf{J}_{\mathtt{ne}}^{(m)}], \ \mathbf{J}_{\mathtt{1}\delta}^{(m)} = \frac{\partial \mathbf{g}}{\partial \boldsymbol{\mu}_{\mathtt{1}\delta}}\Big|_{\boldsymbol{\mu}_{\mathtt{sxe}}^{(m)}, \boldsymbol{\mu}_{\mathtt{1}}, \boldsymbol{\mu}_{\mathtt{n}}}$$

and α is used to improve the stability of noise estimation. Note that the above updating formula only considers the static parameters in the auxiliary function. It is possible to include all the compensated parameters. A good initialisation of noise parameters is important. In [1], the standard VTS-based noise estimation and a guess of T_{60} value were used to initialise the reverberant and additive noise. This is also adopted in this work. During the noise estimation, it is also important to ensure that every update increases the auxiliary. More details about the noise estimation is given in [1].

3. Reverberant adaptive training

The above noise estimation assumes that the acoustic model is trained from clean data. The adaptive training framework can be applied in which both the acoustic model and the noise model

are trained in a full ML framework on multi-conditional data. This is a powerful technique to factor out the unwanted acoustic factors, such as speaker differences and noise distortions, yielding a canonical model $\mathcal{M}_c$ that models only the relevant phoneme variations. The adaptive training framework is extended in this work to handle to reverberant and additive noise distortions. This is referred to as reverberant adaptive training (RAT).

In adaptive training, both the canonical model $\mathcal{M}_c$ and a set of noise models $\mathbf{\Phi}$ are iteratively estimated using EM. First, give the current canonical model, the noise models $\mathbf{\Phi}$ are estimated for each utterance[1], then the canonical model $\mathcal{M}_c$ is updated given the current noise models. Multiple iterations may be performed to interleave optimisation in the EM framework. With RAT, the following auxiliary is used:

$$\mathcal{Q}(\mathcal{M}_c, \{\mathbf{\Phi}^{(u)}\}) = \sum_{u,t,m} \gamma_t^{(mu)} \log p(\boldsymbol{z}_t^{(u)}; \hat{\boldsymbol{\mu}}_{\mathsf{z}}^{(mu)}, \hat{\mathbf{\Sigma}}_{\mathsf{z}}^{(mu)}) \quad (13)$$

where u is the index of utterance. For example, $\gamma_t^{(mu)}$ is the posterior of component m at time t for the u-th utterance.

Given the canonical model, estimating the reverberant and additive noise parameter is described in section 2.2. After the noise parameter updated, the canonical model parameter is retrained. As in RVTSJ adaptation, only the mean is compensated for reverberation, in RAT, only the extended mean $\overline{\boldsymbol{\mu}}_{\mathsf{x}}^{(m)}$ will be updated. The auxiliary function, where only terms dependent on the $\overline{\boldsymbol{\mu}}_{\mathsf{x}}^{(m)}$ are shown, is:

$$\mathcal{Q}(\hat{\mathcal{M}}_c; \mathcal{M}_c) = -\frac{1}{2} \sum_u \sum_m \gamma^{(mu)} \hat{\boldsymbol{\mu}}_{\mathsf{z}}^{(mu)\mathsf{T}} \cdot \mathbf{\Sigma}_{\mathsf{z}}^{(mu)-1} \left(\hat{\boldsymbol{\mu}}_{\mathsf{z}}^{(mu)} - 2\mathbf{\Gamma}_{\mathsf{z}}^{(mu)} \right) \quad (14)$$

where $\gamma^{(mu)} = \sum_t \gamma_t^{(mu)}$ and $\mathbf{\Gamma}_{\mathsf{z}}^{(mu)} = \frac{1}{\gamma^{(mu)}} \sum_t \gamma_t^{(mu)} \boldsymbol{z}_t$ Similar as in the noise estimation, the VTS is applied to expand $\boldsymbol{\mu}_{\mathsf{sz}}^{(m)}$ using the current canonical model estimates, i.e.,

$$\hat{\boldsymbol{\mu}}_{\mathsf{z}}^{(m)} \approx \begin{bmatrix} \boldsymbol{\mu}_{\mathsf{sz}}^{(m)} \\ \boldsymbol{\mu}_{\Delta\mathsf{z}}^{(m)} \end{bmatrix} + \begin{bmatrix} \mathbf{J}_{\mathsf{xe}}^{(mu)} & \mathbf{0} \\ \mathbf{0} & \mathbf{J}_{\mathsf{xe}}^{(mu)} \end{bmatrix} \begin{bmatrix} \mathbf{P}_{\mathsf{s}} \\ \mathbf{P}_{\Delta} \end{bmatrix} (\hat{\overline{\boldsymbol{\mu}}}_{\mathsf{x}}^{(m)} - \overline{\boldsymbol{\mu}}_{\mathsf{x}}^{(m)}) \quad (15)$$

Differentiating the auxiliary function and equating to zero gives the following update:

$$\hat{\overline{\boldsymbol{\mu}}}_{\mathsf{x}}^{(m)} = \overline{\boldsymbol{\mu}}_{\mathsf{x}}^{(m)} + \zeta \left(\sum_u \gamma^{(mu)} \mathbf{K}^{(mu)\mathsf{T}} \mathbf{\Sigma}_{\mathsf{z}}^{(mu)-1} \mathbf{K}^{(mu)\mathsf{T}} + \beta \mathbf{I} \right)^{-1} \times \left(\sum_u \gamma^{(mu)} \mathbf{K}^{(mu)\mathsf{T}} \mathbf{\Sigma}_{\mathsf{z}}^{(mu)-1} (\mathbf{\Gamma}_{\mathsf{z}}^{(mu)} - \boldsymbol{\mu}_{\mathsf{z}}^{(mu)}) \right) \quad (16)$$

where $\mathbf{K}^{(mu)} = \begin{bmatrix} \mathbf{J}_{\mathsf{xe}}^{(mu)} \mathbf{P}_{\mathsf{s}} \\ \mathbf{J}_{\mathsf{xe}}^{(mu)} \mathbf{P}_{\Delta} \end{bmatrix}$, β is a parameter used to stabilise the canonical model parameter update, and ζ is the step size. For every update, the step size ζ is set as 1 initially. Due to the approximation made in Eq. (15), it is necessary to check the auxiliary function after every update to ensure the auxiliary is increasing. If not, a simple back-off procedure, similar to the one used in [2], is used to reduce the step size until the auxiliary increases. Since the auxiliary function of canonical model involves all the utterance, to make this back-off procedure possible, it is necessary to save $\mathbf{\Gamma}_{\mathsf{z}}^{(mu)}$ for each component m and each utterance u, provided $\gamma^{(mu)}$ is not zero, This is impractical for media/large vocabulary task. An approximation of the auxiliary function in Eq. (14) is used: for each component update, the summation over u is done on a subset $\mathcal{U}_m = \{u | \gamma^{(mu)} \geq \theta_m\}$, where θ_m is choose such that the top N utterances are in this subset. $N = 156$ is used in this work and it was found this yields good increase in likelihood.

Since the auxiliary function is highly non-linear, it is crucial to have a good initialisation of the canonical model parameters. In this work, the following strategy is used: the standard model parameters, $\boldsymbol{\mu}_{\mathsf{x}}^{(m)}, \mathbf{\Sigma}_{\mathsf{x}}^{(m)}$ are set as the parameter obtained by the VTS-based adaptive training (VAT), whilst for $\overline{\boldsymbol{\mu}}_{\mathsf{x}}^{(m)}$, it is assumed that $\boldsymbol{\mu}_{\mathsf{x}\delta}^{(m)}, \delta = 0 \ldots n$ is a smooth trajectory starting from $\boldsymbol{\mu}_{\mathsf{x}0}^{(m)} = \boldsymbol{\mu}_{\mathsf{x}}^{(m)}$; hence the reconstruction error is minimised. This amounts to the following optimisation problem :

$$\begin{aligned} \min \quad & \sum_{\delta=1}^{n} w_\delta \| \mathbf{Q}_\delta \overline{\boldsymbol{\mu}}_{\mathsf{x}}^{(m)} - \boldsymbol{\mu}_{\mathsf{x}}^{(m)} \|^2 \\ \text{s.t.} \quad & \mathbf{Q}_0 \overline{\boldsymbol{\mu}}_{\mathsf{x}}^{(m)} = \boldsymbol{\mu}_{\mathsf{x}}^{(m)} \end{aligned} \quad (17)$$

where $\mathbf{Q}_\delta$ is the matrix that maps $\overline{\boldsymbol{\mu}}_{\mathsf{x}}^{(m)}$ to $\boldsymbol{\mu}_{\mathsf{x}\delta}^{(m)}$, $w_\delta = 10^{-\delta \frac{3\Delta}{T_{60}}}$, Δ is the shift of the analysis window (10ms), and T_{60} the median of reverberation time in the multi-style training data (400ms in this work).

4. Experiments

A reverberant version of the AURORA4 task was used for evaluation. The original AURORA4 task is derived from Wall Street Journal (WSJ0) 5k-word dictation task. The WSJ0 training set, consisted of 7138 utterances from 83 speakers, recorded by close-talking microphones, were used as the clean training set in the experiments. To create a multi-conditional training set with reverberation and background additive noise, the clean training set was passed through the simulation tool in [9]. Two RIRs, recored in an office environment ("office1") and a living room environment ("liv1") were used to filter the clean training set, with the reverberant time ranging from 200ms to 600ms. 6 types of background noises which were used in AURORA4 task were also added, with the SNR ranging from 10dB to 20dB, matching the configuration in AURORA4 task. For the test sets, the 330 utterance from 8 speakers in the AURORA4 set A were filtered by two RIRs, "office1" and "office2", where the latter was recored in another office environment and not observed during training. For each RIR, there were two background noise conditions, "clean" and "restaurant", where for the latter, the noise from test04 in AURORA4 task were extracted and added to the reverberant signal at the SNR ranging from 5dB to 15dB. Note the creation of these sets were different from the method in [1], where the reverberant noise was added after background noise distortion.

The HTK frontend was used to derive a 39-dimensional feature vector, consisting of 12 MFCCs, extracted from magnitude spectrum, appended with zeroth cepstrum, delta and delta-delta coefficients. A Cross-word triphone model with 3140 tied states and 16 components per state was built. This model topology was used for all the acoustic models throughout the experiments. For the extended model statistics $\overline{\boldsymbol{\mu}}_{\mathsf{x}}^{(m)}$, the feature vector was appended with high-order DCT elements of an appropriate window width. $n = 10, w = 4$ were used as the length of history frames and the window length used for calculating the dynamic parameters, respectively. The standard bi-gram LM for the AURORA4 task was used in decoding. All the adaptation in this work were performed in a unsupervised mode and the noise models were all estimated at the utterance level.

Experiments were first run using the clean-trained acoustic

[1] It is assumed in this work, each utterance has a unique noise condition, thus a homogeneous block.

noise condition		adaptation		
rev.	add.	—	VTS	RVTSJ
—	clean	7.1	6.9	7.3
	rest.	60.0	19.5	20.0
office1	clean	70.3	43.7	25.4
	rest.	97.3	51.6	43.1
office2	clean	60.1	30.9	16.5
	rest.	97.6	48.8	47.1

Table 1: Performance (in WER %) of the clean-trained acoustic model operated in a noisy and/or reverberant environment.

model. VTS-based noise model parameter was first estimated using multiple EM iterations. The acoustic model was then compensated and used to generate supervision hypothesis. This supervision hypothesis was used for updating VTS noise model and estimating the RVTSJ noise model as well. For comparison, experiments were also run on the 01(clean condition) and 04(restaurant noise, no reverberation) sets in the AURORA4 task. Results are shown in Table 1. As expected, the clean-trained acoustic model is quite fragile to the environment: its performance were greatly impacted by the additive and/or the reverberant noise while reverberation seems to be a more detrimental factor than additive noise. Recognition in a noisy and reverberant environment is the most challenging task, as both distortions cause large mismatch between the training and testing data. Performing VTS compensation significantly reduced the mismatch caused by noise and reverberation. RVTSJ adaptation on non-reverberant data gave similar but slightly worse performance (less than 0.5% absolute degradation). This is felt be a limitation of current noise estimation method when the reverberant noise ($\boldsymbol{\mu}_{1\delta}, \delta = 1 \ldots n$) approaches to $-\infty$. However, when the reverberation is presented in the data, RVTSJ gave large gains over VTS. This demonstrated that RVTSJ is modelling the impact of reverberation, which is not modelled well by VTS.

In the second set of experiments, multi-conditional training data was used to build acoustic models. Firstly, as in [6], stereo data were used to build a MST system. Starting from this MST model, VAT system was build, which in turn severs as an initialisation for the RAT to begin. The parameter β was reduced from 8 to 1 for 4 iterations of RAT canonical model re-estimation. This was followed by 2 iterations of noise model update. This process is repeated one more time to yield the final RAT model. An initial decoding using MST system without adaptation was run. Results are shown in line 2, Table 1. Compared with performance in Table 1, in the environment observed during training, "office1", MST model works quite well, producing better performance than adapting clean-trained acoustic models to the target environment. However, when it was operated in "office2", an environment not observed during training , performances were degraded. This demonstrates multi-style training introduces biased toward the training environments while not generalised well to other unseen conditions. The MST system was also adapted by an CMLLR transform for each speaker. As a general adaptation scheme, CMLLR is powerful to reduce the mismatch at testing. As shown in line 3, Table 2, this yields large error reduction in all the environments. This was consistent with the finding in [6, 5]. The hypothesis obtained by MST+CMLLR will be used as supervisions for the following VTS/RVTSJ adaptation experiments. Compared with the CMLLR adapted MST system, VAT system with VTS adaptation was worse performed when there is only reverberation distortion, but gave gains when the additive noise is also presented in

Systems	Adaptation	office1		office2		Avg.
		clean	rest.	clean	rest.	
MST	—	19.9	38.3	37.4	63.6	39.8
	CMLLR	14.1	30.1	14.7	48.4	26.8
VAT	VTS	15.1	29.0	18.7	44.5	26.8
	RVTSJ	14.9	29.6	18.3	43.6	26.6
RAT	RVTSJ	13.7	28.5	15.0	42.2	24.9

Table 2: Performance of multi-style trained and adaptively trained acoustic models in reverberant environments.

the environment. This is due to that VTS is designed to compensate the impact of additive noise, while CMLLR can be used for general adaption. Based on the VAT canonical model, the extended model statistics $\{\overline{\boldsymbol{\mu}}_{\mathbf{x}}^{(m)}\}$ were initialised by solving the optimisation in Eq. (17). Given the extended model statistics, RVTSJ adaptation of VAT was performed, which gave small gains in average (0.2%). RVTSJ adaptation of the RAT system further improves the performance. Compared with VAT system, RAT yields 0.5% to 1.3% absolute gains for the office1 environment, and 2.3% to 3.7% for the office2 environment. This demonstrates that RVTSJ adaptation models the impact of both the reverberant and additive noise, while RAT produces a canonical model neutral to these distortions to some extent. On average, RAT system gave the best performance, yielding 1.9% absolute gains over CMLLR adapted MST system.

5. Conclusions

This work investigates model based approaches to adaptive training in reverberant environments. A new adaptive training algorithm, reverberant adaptive training (RAT), is proposed, where the RVTSJ adaptation is used in both testing and training stage. An ML estimation of canonical model parameters in the EM framework is presented. Experiments conducted on a reverberant simulated AURORA4 task demonstrates RAT produces a canonical model neutral (to some extent) to the reverberant and additive noise variations and provides better results than adapting the multi-style trained acoustic model using CMLLR.

6. References

[1] Y.-Q. Wang and M. J. F. Gales, "Improving reverberant VTS for hands-free robust speech recognition," in *Proc. ASRU-2011*.

[2] H. Liao and M. J. F. Gales, "Joint uncertainty decoding for robust large vocabulary speech recognition," Tech. Rep. CUED/F-INFENG/TR552, University of Cambridge, 2006.

[3] J. Li, D. Yu, Y. Gong, and A. Acero, "High-performance HMM adaptation with joint compensation of additive and convolutive distortions via vector Taylor series," in *Proc. ASRU-2007*.

[4] M. J. F. Gales and Y.-Q. Wang, "Model-based approaches to handling additive noise in reverberant environments," in *Workshop on Hands-free Speech Communication and Microphone Arrays*, 2011.

[5] M. Matassoni et al., "Hidden Markov model training with contaminated speech material for distant-talking speech recognition," *Computer Speech & Language*, vol. 16, no. 2, pp. 205–223, 2002.

[6] A. Sehr et al., "Multi-style training of HMMs with stereo data for reverberation-robust speech recognition," in *Workshop on Hands-free Speech Communication and Microphone Arrays*, 2011.

[7] T. Anastasakos, J. McDonough, R. Schwartz, and J. Makhoul, "A compact model for speaker adaptive training," in *Proc. ICSLP-96*, pp. 1137–1140.

[8] O. Kalinli, M. L. Seltzer, and A. Acero, "Noise adaptive training using a vector Taylor series approach for noise robust automatic speech recognition," in *Proc. ICASSP-2009*.

[9] H. G. Hirsch and H. Finster, "The simulation of realistic acoustic input scenarios for speech recognition systems," in *Proc. ICSLP*, 2005.