Minimum Phone Error Training of Precision Matrix Models

Khe Chai Sim, Student Member, IEEE and M.J.F. Gales, Member, IEEE

Abstract—Gaussian Mixture Models (GMMs) are commonly used as the output density function for Large Vocabulary Continuous Speech Recognition (LVCSR) systems. A standard problem when using multivariate GMMs to classify data is how to accurately represent the correlations in the feature vector. Full covariance matrices yield a good model, but dramatically increase the number of model parameters. Hence diagonal covariance matrices are commonly used. Structured precision matrix approximations provide an alternative, flexible and compact representation. Schemes in this category include the extended maximum likelihood linear transform and subspace for precision and mean models. This paper examines how these precision matrix models can be discriminatively trained and used on state-of-the-art speech recognition tasks. In particular the use of the minimum phone error criterion is investigated. Implementation issues associated with building LVCSR systems are also addressed. These models are evaluated and compared using large vocabulary continuous telephone speech and broadcast news English tasks.

Index Terms—precision matrix modelling, minimum phone error, discriminative training, large vocabulary continuous speech recognition.

I. INTRODUCTION

STATE-OF-THE-ART speech recognition systems are typically based on continuous density hidden Markov models [1] with Gaussian Mixture Models (GMMs) representing the output distribution associated with each state. A standard problem when using multivariate GMMs to classify data is how to accurately model the correlations in the feature vector. The use of a full covariance matrix for each Gaussian component dominates the total number of model parameters and dramatically increases the computational cost to train and perform recognition with these models. Furthermore, a large amount of training data is required to ensure robust model estimation. For these reasons, more compact and efficient correlation modelling techniques are required, particularly for a Large Vocabulary Continuous Speech Recognition (LVCSR) [2] system, which comprises many Gaussian components (typically greater than 100,000) and high dimensional data (typically 39 or 52). The conventional approach to addressing these problems is to use a diagonal covariance matrix approximation. The feature dimensions are assumed to be uncorrelated given a particular component. Several methods have been employed to improve the validity of this assumption. For example, the use of Mel Frequency Cepstral Coefficients (MFCC) [3] and Perceptual Linear Prediction (PLP) [4] coefficients provide data with low correlation. Further decorrelation can be achieved using feature transformation techniques such as Linear Discriminant Analysis (LDA) [5], Heteroscedastic LDA (HLDA) [6] and Heteroscedastic Discriminant Analysis (HDA) [7].

Recently, more advanced covariance modelling techniques have been found to give improvements over the feature decorrelating schemes above. Techniques that approximate the inverse covariance (precision) matrices are commonly used. This is more efficient than modelling the covariance matrix, as it eliminates the need to invert the covariance matrices as required by schemes such as the Factor-Analysed HMMs [8]. This yields efficient likelihood computation for precision matrix models. Examples of these models are the Semi-Tied Covariance (STC) [9], Extended MLLT (EMLLT) [10] and Subspace for Precision And Mean (SPAM) [11] models. These models have been successfully applied to LVCSR systems using the Maximum Likelihood (ML) training scheme [9], [12], [13].

For many years, ML estimation has been the standard approach to train the HMMs for speech recognition. However, discriminative training has been found to yield promising gain over the ML training on diagonal covariance matrix systems [14], [15]. This has motivated the use of discriminative training for many state-of-the-art LVCSR systems [16], [17]. The STC [18] and SPAM [19] models have previously been discriminatively trained using the Maximum Mutual Information (MMI) criterion on small and medium vocabulary systems. An alternative discriminative training criterion, Minimum Phone Error (MPE), has been found to consistently outperform MMI training on large vocabulary diagonal covariance matrix systems [15]. This paper investigates the use of MPE trained precision matrix models for LVCSR systems. The MPE training approach adopted in this paper is based on the optimisation of the weak-sense auxiliary function with I-smoothing, as presented in [15]. Implementation issues regarding building LVCSR systems with precision matrix models will also be discussed.

This paper is organised as follows: Section II describes a generic framework of basis superposition [20], [21] which subsumes various forms of precision matrix modelling techniques. Next, discriminative training of precision matrix models based on the MPE criterion will be discussed in Section III. Section IV then addresses the implementation issues of these precision matrix models for LVCSR systems. Experimental results on Conversational Telephone Speech (CTS) and Broadcast News (BN) English tasks are presented in Section V.

II. PRECISION MATRIX MODELLING

Compact precision matrix modelling has been found to yield good gains over the diagonal covariance matrix approximation for GMM covariance modelling. The generic framework of basis superposition [21] may be used as a convenient way of
analysing various forms of precision matrix models, such as the STC, EMLLT and SPAM\(^1\) models. Within this framework, the precision matrix, \(P_m\), of the Gaussian component \(m\) is given by the following general expression

\[
P_m = \sum_{i=1}^{n} \lambda_{im}^{(m)} S_i = \sum_{i=1}^{n} \sum_{r=1}^{R} \lambda_{ir} a_{ir} \lambda_{ir}^T a_{ir}
\]  

(1)

where \(n\) is the number of basis (basis order), \(S_i\) is a set of symmetric basis matrices and \(\lambda_{im}^{(m)}\) are the corresponding superposition coefficients for component \(m\). The basis matrices, \(S_i\), can be further decomposed into a linear combination of \(R\) basis row vectors, \(a_{ir}\) weighted by \(\lambda_{ir}\) and \(R\) denotes the rank of \(S_i\). If \(R = 1\), the precision matrix in equation (1) becomes a STC [9] when \(n = d\) and an EMLLT [10] model when \(d < n \leq \frac{d}{2}(d+1)\), where \(d\) is the feature dimensionality. Alternatively, a SPAM [19] model may be modelled with \(R = d\). In this case, provided one of the \(S_i\) is positive-definite, \(n\) is allowed to be as small as 1. Furthermore, setting \(R < d\) yields the Hybrid-EMLLT model [22]. Due to the parameterisation of basis superposition into the global (basis vectors/matrices) and component (basis coefficients) parameters, compact model representation may be achieved via sharing of the basis vectors or matrices. The HLDA precision matrix model (HLDA-PMM) [20] employs tying of the basis coefficients corresponding to the nuisance dimensions, which further reduces the number of model parameters.

One of the attractive attributes of precision matrix modelling is its efficiency during decoding. This can be seen clearly from the log likelihood expression given by

\[
\mathcal{L}(\mu_m, P_m | o_t) = K + \frac{1}{2} \left\{ \log |P_m| - \Omega_{m1} P_m o_m o_m^T \right\}
\]  

(2)

where \(\mathcal{L}(\mu_m, P_m | o_t)\) is the log likelihood of the observation, \(o_t\), given the model parameters of Gaussian component \(m\) and \(o_m = (o_t - \mu_m)\). The constant \(K\) subsumes terms independent of the model parameters, \(\mu_m\) and \(P_m\) are the mean vector and precision matrix of component \(m\) respectively. Modelling the precision matrix, \(P_m\), as a superposition of basis eliminates the need to invert the covariance matrix when computing the likelihood. Furthermore, it is shown in [21] that the terms in equation (2) can be divided into model and observation dependent. The former can be precomputed and cached once the model parameters are loaded. The latter can be cached for each observation and then reused for all the Gaussian components. This yields a significantly cheaper computational cost, which is linearly proportional to the basis order, \(n\).

Maximum Likelihood Estimation (MLE) is a standard approach to finding model parameters. Within the HMM framework, this is commonly optimised using the well-known Baum-Welch (or more generally Expectation Maximisation) [23] algorithm. The auxiliary function to be maximised in the M-step [21] is given by

\[
Q_{m1}(\theta, \theta) = K + \frac{1}{2} \sum_{m=1}^{M} \beta_{m1} \left\{ \log |P_m| - \text{Tr}(P_m \mathbf{W}_{m}^{n1}) \right\}
\]  

(3)

where

\[
\text{Tr}(P_m \mathbf{W}_{m}^{n1}) = \sum_{t=1}^{T} \gamma_{m1}^{(m)}(t)(\theta_t - \mu_m)^T P_m (\theta_t - \mu_m)
\]  

(4)

and

\[
\mathbf{W}_{m}^{n1} = \sum_{t=1}^{T} \gamma_{m1}^{(m)}(t)
\]  

(5)

\[
\beta_{m1} = \sum_{t=1}^{T} \gamma_{m1}^{(m)}(t)
\]  

(6)

\(\mathbf{W}_{m}^{n1}\) and \(\beta_{m1}\) are the ML full covariance statistics and the component occupancy counts respectively. For all the forms of precision matrix modelling considered here, the mean vectors are unconstrained. Thus, the following standard update formulae may be used

\[
\mu_m = \frac{1}{\beta_{m1}} \sum_{t=1}^{T} \gamma_{m1}^{(m)}(t) \theta_t
\]  

(7)

The ML update formulae for various precision matrix models are summarised in [21]. Further details regarding these models may also be obtained from the corresponding literatures (9, 10, 11, 20).

III. MINIMUM PHONE ERROR (MPE) TRAINING

Discriminative training has been found to yield improved performance in LVCSR compared to the conventional ML training [14]. Various forms of discriminative objective functions have been described in these literatures, for example Maximum Mutual Information (MMI), Minimum Phone Error (MPE) and Minimum Word Error (MWE) criteria [14], [15]. Several forms of MMI trained precision matrix models have recently been published. Goel et al., 2003 [19] presented the MMI estimation of the SPAM models with small vocabulary system. McDonough et al. [18] also employed MMI trained STC models in speaker-adapted training (SAT). Tsakalidis et al. [24] introduced Discriminative Likelihood Linear Transform (DLLT), a variant of MLLT whose parameters estimation is also based on the MMI criterion. The consistent improvement of MPE training on large scale diagonal covariance matrix systems compared to the MMI discriminative criterion [15] motivates the investigation of MPE training of precision matrix models on LVCSR systems.

A. Maximising the MPE Objective Function

MPE training aims to minimise the phone classification error (or maximising the phone accuracy). The objective function to be maximised by the MPE training, \(R_{\text{MPE}}(\theta)\), may be expressed as

\[
R_{\text{MPE}}(\theta) = \sum_{r} \sum_{s} p_{w}(O_{r} | s) P(s) \text{PhoneAcc}(s, s_r) \sum_{u} p_{w}(O_{r} | u) P(u)
\]  

(8)
where \( O_r \) is the \( r \)th training sentence and \( P(s) \) is the language model probability for sentence \( s \). \( \kappa \) is an acoustic de-weighting factor, which can be adjusted to improve the test-set performance. PhoneAcc(\( s, s_r \)) represent the raw phone accuracies of the sentence \( s \) given the correct sentence \( s_r \).

The MPE objective function is difficult to optimise directly. In this paper, MPE training of the precision matrix models is based on the approach presented by Povey et al. [15], [25]. The MPE objective function (8) is maximised using an auxiliary function of the form2

\[
\mathcal{Q}^n(\theta, \hat{\theta}) = \mathcal{Q}^n(\theta, \hat{\theta}) - \mathcal{Q}^d(\theta, \hat{\theta}) + \mathcal{F}(\theta, \hat{\theta})
\]  

(9)

where \( \mathcal{Q}^n(\theta, \hat{\theta}) \) and \( \mathcal{Q}^d(\theta, \hat{\theta}) \) are the numerator and denominator terms respectively, which take the same form as the ML auxiliary function in equation (3), except that the statistics are now derived based on the numerator and denominator ‘counts’, \( \gamma_m(t) \) and \( \gamma^n_m(t) \) respectively (cf. equations (5) and (6)). The numerator (denominator) counts are computed by adding the positive (negative) terms of the differential of the MPE objective function with respect to the log likelihood of all the phone arcs at time \( t \) [15]. \( \mathcal{F}(\theta, \hat{\theta}) \) is a smoothing function which, as suggested in [15], takes the form

\[
\mathcal{F}(\theta, \hat{\theta}) = K + \frac{1}{2} \sum_{m=1}^{M} \log |P_m| - \text{Tr}(P_m \Sigma_m)
\]  

(10)

where \( \Sigma_m \) is the current estimate of the full covariance matrix and \( D_m \) is a component-dependent constant that controls the amount of \( \Sigma_m \) to be smoothed onto the covariance statistics. Equation (9) is referred to as the weak-sense auxiliary function in [15], [25] because an increase in this function does not guarantee an increase in the objective function. In the following, the sufficient statistics required to optimise this weak-sense auxiliary function will be discussed and model parameter update formulae for the EMILLT and SPAM models will be given.

B. Sufficient Statistics for MPE Training

The full ML covariance statistics, \( \mathbf{W}^{n1} \), can be rewritten in terms of the sufficient statistics such that

\[
\mathbf{W}^{n1} = \frac{\mathbf{Y}^{n1} - \mathbf{x}^{n1} \mu^{n1} - \mu^{n1} \mathbf{x}^{n1} + \beta_m^{n1} \mu^{n1} \mu^{n1}}{\beta_m^{n1}}
\]  

(11)

where the sufficient statistics, \( \Theta^{n1} = \{\beta_m^{n1}, \mathbf{x}^{n1}, \mathbf{Y}^{n1}\} \) for all components \( m \), are given by equation (6),

\[
\mathbf{x}^{n1} = \frac{T}{\sum_{t=1}^{T} \gamma^{n1}_m(t)} \mathbf{o}_t \quad \mathbf{Y}^{n1} = \frac{T}{\sum_{t=1}^{T} \gamma^{n1}_m(t)} \mathbf{o}_t \mathbf{d}_t
\]

Given the set of parameters, \( \theta \), the ML auxiliary function (3) can be rewritten in terms of the ML statistics \( \Theta^{n1} \)

\[
\mathcal{Q}^{n1}(\theta, \hat{\theta}) = \mathcal{G}(\Theta^{n1})
\]  

(12)

where

\[
\mathcal{G}(\Theta^{n1}) = K + \frac{1}{2} \sum_{m=1}^{M} \beta_m^{n1} \log |P_m| - \text{Tr}(P_m \mathbf{W}^{n1}_m) \]

(13)

Equation (9) can also be expressed in terms of sufficient statistics

\[
\mathcal{Q}(\theta, \hat{\theta}) = \mathcal{G}(\Theta^c) - \mathcal{G}(\Theta^d) + \mathcal{G}(\Theta^{n1})
\]  

(14)

using the function \( \mathcal{G}(\cdot) \) defined in equation (13). \( \Theta^c \) and \( \Theta^d \) denote the sufficient statistics for numerator and denominator respectively. The set of parameters, \( \Theta^{n1} \), which correspond to the smoothing function (10), \( \mathcal{F}(\theta, \hat{\theta}) = \mathcal{G}(\Theta^{n1}) \), are given by

\[
\beta_m^{n1} = D_m
\]  

(15)

\[
\mathbf{x}^{n1}_m = D_m \mu_m
\]  

(16)

\[
\mathbf{Y}^{n1}_m = D_m (\Sigma_m + \mu_m \mu^t_m)
\]  

(17)

Maximising this auxiliary function with respect to the mean and covariance matrix parameters yields the following update formulae

\[
\mu_m = \frac{\mathbf{x}^{n1}_m - \mathbf{x}^d_m + D_m \mu^t_m}{\beta_m^{n1} - \beta_m^d + D_m}
\]  

(18)

\[
\mathbf{W}^{npe}_m = \frac{\mathbf{Y}^{n1}_m - \mathbf{y}^d + D_m (\Sigma_m + \mu_m \mu^t_m) - \mu_m \mu^t_m}{\beta_m^{n1} - \beta_m^d + D_m}
\]  

(19)

It is also possible to consider a set of combined statistics

\[
\Theta^c = \Theta^c - \Theta^d
\]  

(20)

where this set “¬” operator yields \( \beta_m^c = \beta_m^c - \beta_m^d \) and similarly for \( \mathbf{Y}_m^c \) and \( \mathbf{x}^c_m \). Using this concept of functions over statistics it is simple to incorporate smoothing techniques such as I-smoothing [15] and Maximum a-Posteriori (MAP) [25] smoothing. To ensure that the auxiliary function is valid, \( \mathbf{W}^{npe}_m \) is required to be positive-definite. Combining equations (18) and (19) gives the full covariance statistics in terms of \( D_m \).

\[
\mathbf{W}^{npe}_m = \frac{B_2 \mathbf{D}^2_m + B_1 \mathbf{D}_m + B_0}{\beta_m^e + D_m}
\]  

(21)

where

\[
B_2 = \Sigma_m
\]

(22)

\[
B_1 = \mathbf{Y}_m^c + \beta_m^e (\Sigma_m + \mu_m \mu^t_m) - \mu_m \mu^t_m
\]

(23)

\[
B_0 = \beta_m^e \mathbf{y}^c_m - \mathbf{x}^c_m \mu^t_m
\]

(24)

The constant, \( D_m \) is used to ensure the positive-definiteness of the resulting precision matrices. In general, this is equivalent to imposing positive-definite constraint on \( \mathbf{W}^{npe}_m \), which is given by the largest positive eigenvalues of the Quadratic Eigenvalue Problem (QEP) of equation (21) [19]. However, Section IV-E will show that solving the QEP is unnecessary if one intends to update only the basis coefficients. Furthermore, a lower bound is also applied to the smoothing constant value such that the actual smoothing constant value, \( D_m \), is given by

\[
D_m = \max (2D_m, E/\beta_m^e)
\]  

(25)
where the lower bound $E_{\theta_m}^c$ is applied to ensure that the combined occupancy count, $\beta_m^c$, is greater than zero. $E = 2$ is empirically found to lead to good test-set performance [15].

Once the overall statistics in equations (18) and (19) are found, the auxiliary function in equation (9) can be maximised to discriminatively train the precision matrix model parameters. This function is exactly the same as the ML auxiliary function. Thus, all the standard ML optimisation formulae described in [21] may be used by replacing the ML statistics with the combined MPE ones:

- **STC/EMLLT Basis coefficient update:**

$$\hat{\lambda}_m^{(m)} = \hat{\lambda}_m^{(m)} + \left( \frac{1}{a_i \mathbf{W}_m^{\text{mpe}} a_i} - \frac{1}{a_i \Sigma_m a_i} \right)$$

where $\hat{\lambda}_m^{(m)}$ and $\Sigma_m$ are the current estimates of the basis coefficient and full covariance matrix respectively.

- **SPAM Basis coefficient update:** A simple line search along the Polak-Ribiere conjugate-gradient [28], $d^m$, is employed to maximise the change in auxiliary function:

$$Q_m^\theta = \frac{r_m^\text{mpe}}{2} \left\{ \sum_{i=1}^d \log \left( 1 + \Delta_m^i \right) - \Delta_m \sum_{i=1}^n d_m^i \right\}$$

where $\Delta_m$ is the step size along $d^m$ that maximises $Q_m^\theta$, $d_m^i$ is the $i$th element of $d^m$, $z_m^i$ is the $i$th eigenvalue of $\mathbf{P}_m$, $(\sum_{i=1}^d d_m^i S_i) \mathbf{P}_m^{-1}$ and $\mathbf{w}_m^{\text{mpe}} = \text{Tr}(\mathbf{W}_m^{\text{mpe}} S_i)$ is the projected statistics (see Section IV-A).

### C. I-Smoothing

I-smoothing is an interpolation technique proposed by Povey et al. [15] that incorporates prior information over the Gaussian parameters to control the convergence of the MPE training process. The prior is based on the ML statistics. Using I-smoothing requires the redefinition of the weak-sense auxiliary function (9) as

$$Q(\theta, \hat{\theta}) = Q^\theta(\theta, \hat{\theta}) - Q^d(\theta, \hat{\theta}) + \mathcal{F}(\theta, \hat{\theta}) + \log(p(\theta))$$

where

$$\log(p(\theta)) = K + \frac{\tau_f}{2} \sum_{m=1}^M \left\{ \log |\mathbf{P}_m| - \text{Tr}(\mathbf{P}_m \mathbf{W}_m^{\text{m1}}) \right\}$$

and $\mathbf{W}_m^{\text{m1}}$ is given by equation (11). $\tau_f$ is the I-smoothing constant. The prior can be regarded as the log likelihood of $\tau_f$ data points with the mean and variance of the ML estimate. Incorporating I-smoothing is easy by rewriting the combined statistics as

$$\mathbf{x}_m^c = \mathbf{x}_m^c - \mathbf{x}_m^d + \frac{\tau_f}{\beta_m^c} \mathbf{x}_m^m$$

$$\mathbf{y}_m^c = \mathbf{y}_m^c - \mathbf{y}_m^d + \frac{\tau_f}{\beta_m^c} \mathbf{y}_m^m$$

$$\beta_m^c = \beta_m^d + \tau_f$$

It is simple to see that as the I-smoothing constant, $\tau_f$, tends to infinity, the resulting estimation formulae tend to those of the ML training.

### IV. IMPLEMENTATION ISSUES

This section addresses the implementation issues of various precision matrix models, paying particular attention to building LVCSR systems. Many of these models have been successfully applied to LVCSR systems [9], [12], [13]. This paper emphasises issues such as memory requirement, computational feasibility and training robustness in LVCSR systems. System efficiency may be adversely affected if these issues are not addressed properly.

#### A. Memory Issues

One issue with implementing precision matrix models on LVCSR systems is the large amount of memory requirement for full covarance statistics accumulation. However, given a good set of basis (through good initialisation schemes discussed in Section IV-B or a few ML training iterations), the precision matrix models can be refined by simply updating the basis coefficients alone. This is more efficient in terms of memory requirement because the sufficient statistics can be reduced to a more compact form known as the projected statistics, $\mathbf{w}_m^{\text{mpe}}$, where

$$\mathbf{w}_m^{\text{mpe}} = \text{Tr}(\mathbf{S}_i \mathbf{W}_m^{\text{mpe}})$$

$$= \frac{1}{\beta_m} \frac{D_m^2 + b_1^{\text{mpe}}}{D_m^2 + b_0^{\text{mpe}}} \right) \beta_m + D_m$$


#### B. Basis Initialisations

In the basis superposition framework, the basis vectors or matrices extract the common structure of the precision matrices of all Gaussian components. The update of the basis vectors for EMLLT models and basis matrices for SPAM models does not have a closed form solution and generic optimisation routines such as the conjugate gradient decent method [28] have to be used. Thus, it is important to obtain a good initial set of basis to allow fast convergence and avoid hitting a poor local maximum during parameters estimation process. This is especially true for the EMLLT and SPAM models, where the update of basis vectors/matrices is slow. For STC, a trivial identity initialisation leads to a diagonal covariance matrix system. Several basis initialisation schemes are available for the EMLLT models [21]. The STC-HLDA initialisation scheme was found to be the best in terms of WER.
performance and is more flexible than simply stacking multiple STC transforms [10], which constrains \( n \) to be a multiple of \( d \).

According to [11], it is useful to initialise the set of basis matrices \( \{ \mathbf{S}_i \} \) for SPAM models as the symmetric matrices associated to the top \( n-1 \) singular vectors of the matrix

\[
\mathbf{V} = \frac{\sum_{m=1}^{M} c_m \mathbf{v}_m \mathbf{v}_m^T}{\sum_{m=1}^{M} c_m}
\]

(31)

where \( c_m \) is the component weight of Gaussian component \( m \) of the GMM distribution and \( \mathbf{v}_m = \text{vec}(\{ \mathbf{W}_{m}^{1/2} \}) \). \( \text{vec}(\mathbf{A}) \) is a row-wise vectorisation of the lower-triangular elements of the symmetric matrix \( \mathbf{A} \). On large systems, full covariance statistics associated with each Gaussian component, \( \mathbf{W}_{m}^{1/2} \), may not be obtained robustly. Instead, it was found that using the inverse of the state-level covariance statistics produced a more reliable set of basis matrices [21]. The remaining basis matrix is initialised as the average precision matrix of the Gaussian components in the system to form a total of \( n \) basis. This is required to ensure that at least one basis matrix is positive-definite to allow a valid initialisation by simply initialising the corresponding basis coefficient to one and the rest zero [11].

C. Variance Flooring

In situations of data sparseness, which is common in LVCSR systems, a variance floor is required to prevent over-fitting. It imposes a lower bound to the variances (diagonal elements of the covariance matrix). The standard form, for example in HTK [29], of the variance floor, \( \sigma_{ii}^{(v)f} \), is

\[
\sigma_{ii}^{(v)f} = \frac{\alpha \sum_{s=1}^{S} \sigma_{ii}^{(s)} \beta_{m}^{(s)}}{\sum_{s=1}^{S} \beta_{m}}
\]

(32)

where \( S \) is the total number of HMM states, \( \sigma_{ii}^{(s)} \) and \( \beta_{m} \) are the state-level variance and occupancy count respectively for state \( s \) (similar to \( \sigma_{ii}^{(m)} \) and \( \beta_{m}^{(s)} \), but summed over all Gaussian components within state \( s \)). \( \alpha \) is a scaling factor is typically set as 0.1. This method may be directly applied to the basis coefficients of the STC models due to the form of basis vectors [21].

The above method is not applicable to EMLLT models due to the existence of negative basis coefficients. Instead, in this work, the variance floor is applied to the full covariance or projected statistics used to update the model parameters [20], [21]. Unfortunately, it is not possible to apply the variance floor onto the projected statistics, \( \text{Tr}(\mathbf{W}_{m}^{\text{proj}} \mathbf{S}_i) \), for SPAM models. However, if one of the basis matrices is initialised to be positive-definite \( (\mathbf{S}_i) \) [11], the coefficient corresponding to \( \mathbf{S}_i \) can be gradually increased until the final precision matrices satisfy the variance floor condition. However, this approach increases the computational load. Thus, no variance floor is applied to SPAM models in this paper.

D. Multiple Transformations Scheme

The basis superposition framework introduced earlier has an extreme basis tying scheme. A single set of basis matrices is shared by all the Gaussian components. This requires a large set of basis matrices to yield good representation. Alternatively, the components can be partitioned into clusters. Each cluster will then contain a smaller number of components. Extracting basis for each cluster of Gaussian components should yield more accurate basis information. The basis matrices are now tied at a cluster level. This leads to a multiple projections scheme where each projection is associated with the set of basis matrices. A good summarisation of multiple projections schemes is given in [30]. Multiple HLDA projections models have been found to lead to good recognition performance [31]. For multiple projections basis superposition models, equation (1) can be rewritten as

\[
P_{m} = \sum_{i=1}^{n} \lambda_{ii}^{(m)} \mathbf{S}_i^{(m)}
\]

(33)

where \( m \in g(m) \) and \( g(m) \) denotes the cluster to which component \( m \) belongs to. There are many ways to perform Gaussian clustering. One way is to use a regression class tree [32] and the terminal nodes of the tree corresponds to the clusters of Gaussian components. This is the approach adopted in this paper.

E. Approximating the Smoothing Constant, \( D_m \)

Determining the smoothing constant value directly as described earlier is memory inefficient. Solving the QEP for equation (21) requires storing of the full covariance statistics [19]. Storing these statistics for large systems results in large memory requirement.

If one is interested only in updating the basis coefficients, imposing the positivity constraint on the projected statistics, \( \mathbf{a}_i \mathbf{W}_{m}^{\text{proj}} \mathbf{a}_j^T \), is sufficient to yield positive-definite precision matrices for STC and EMLLT models provided that the initial precision matrices are positive-definite. This is clearly evident given that equation (26) only depends on the projected statistics. Furthermore, a proof for this is also given in [10] for EMLLT models. Thus, the value of \( D_m \) can be determined by finding the largest positive root of the quadratic equation (30) rather than solving a QEP for equation (21).

Unlike STC and EMLLT models where the basis matrices are rank-1, the ‘projected’ statistics, \( \text{Tr}(\mathbf{W}_{m}^{\text{proj}} \mathbf{S}_i) \), associated with the basis matrices, \( \mathbf{S}_i \) of the SPAM model can not be used to infer the positive-definiteness of the resulting precision matrices. Instead of obtaining the exact smoothing constant value by solving the QEP for equation (21), this value can be approximated by using a pseudo transformation matrix, \( \mathbf{A}^* \). The transformed space is assumed to have negligible correlation such that the QEP is once again broken down into \( n \) independent quadratic equations as for the STC and EMLLT models. Thus, two sets of statistics are required: one for determining the smoothing constant, \( D_m \), and the other one for estimating the model parameters, \( \text{Tr}(\mathbf{W}_{m}^{\text{proj}} \mathbf{S}_j) \). To obtain a good approximation for the smoothing constant, \( \mathbf{A}^* \) should be chosen such that the transformed space is as uncorrelated as possible. It is reasonable to select the STC transform as the pseudo transformation matrix. In the case where STC transform is unavailable, an identity matrix may be used. This was found to be a good approximation [21] and is used in this paper.
V. EXPERIMENTAL RESULTS

Discriminative training of precision matrices was evaluated on an English Conversational Telephone Speech (CTS) task, which consists of multi-speaker spontaneous telephone conversational speech, and an English Broadcast News (BN) task. Both of these are based on data provided by the Linguistic Data Consortium (LDC). Data was coded into 12 PLP coefficients at a frame rate of 10ms with a frame size of 25ms, together with the log energy term, first, second and third derivatives to form 52-dimensional feature vectors. For the CTS task, acoustic models were represented by decision tree state-clustered triphone models with 6189 distinct states. Side-based Central Mean Normalisation (CMN), Cepstral Variance Normalisation (CVN) and Vocal Tract Length Normalisation (VTLN) were also used. Acoustic models for the BN task consist of 7001 distinct states. For the BN task, precision matrix models were trained using only a wide band data. Narrow band speech was decoded using a HLDA MPE diagonal covariance model. Only segment based CMN was used for BN experiments.

The models used in all the experiments were built using the HTK [29]. ML and MPE training were conducted with 4 and 8 iterations respectively. All HLDA [6] systems used a $39 \times 52$ linear transformation matrix\(^3\) trained once using 16 components per state models and fixed for subsequent training. Basis matrices for EMLLT and SPAM models were initialised as described in Section IV-B, where the STC-HLDA method was used for EMLLT models. For memory tractability, only basis coefficients were updated in MPE training. A multi-pass decoding strategy was employed where word lattices were first generated using a bigram language model and a dictionary comprising 58231 words with multiple pronunciation probabilities. These lattices were then rescored using a trigram language model to produce the final 1-best hypotheses.

Initial experiments were conducted based on the h5etrain03 (296 hours) training set and the dev01sub (3 hours) test set of the CTS English task to evaluate the performance of various precision matrix models. The performance of multiple transforms systems was also compared using the same training and test sets. Finally, selected systems were tested on the full CTS (6 hours eval03) and BN (3 hours each for dev03 and eval03) English tasks.

\(^3\)A ML estimated feature projection from the original 52-dimensional space to a 39-dimensional subspace.

A. Development Results

This section discusses the initial evaluation of various precision matrix models. 16-component models were used to allow rapid training for these initial comparisons. The word error rate (WER) numbers are summarised in Table I. The second and third columns show the dimensions for the mean vector and basis coefficients respectively. The HLDA ML model has a WER of 33.5% on dev01sub. If the nuisance dimensions are retained, the equivalent 52 dimensional STC model yields a further 0.2% absolute reduction in WER. By tying the 13 basis coefficients corresponding to the HLDA

<table>
<thead>
<tr>
<th>System</th>
<th>Dimensions</th>
<th>WER (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\mu$</td>
<td>$\Sigma$</td>
</tr>
<tr>
<td>HLDA</td>
<td>39</td>
<td>39.5</td>
</tr>
<tr>
<td>HLDA+EMLLT</td>
<td>39</td>
<td>78.2</td>
</tr>
<tr>
<td>HLDA+SPAM</td>
<td>39</td>
<td>32.0</td>
</tr>
<tr>
<td>HLDA-PMM</td>
<td>39</td>
<td>33.2</td>
</tr>
<tr>
<td>STC</td>
<td>52</td>
<td>52.2</td>
</tr>
<tr>
<td>EMLLT</td>
<td>78</td>
<td>32.6</td>
</tr>
<tr>
<td>SPAM</td>
<td>39</td>
<td>32.8</td>
</tr>
</tbody>
</table>

**TABLE I** COMPARISON OF WER (%) PERFORMANCE OF ML AND MPE TRAINED 16-COMPONENT PRECISION MATRIX MODELS ON dev01sub CTS ENGLISH TASK

nuisance dimensions using a HLDA-PMM model [20], another 0.1% absolute improvement was obtained. With 78 basis coefficients, the EMLLT model is 0.9% absolute better than the HLDA model. The SPAM model, with half the number of basis coefficients, is only 0.2% absolute behind the EMLLT model. Precision matrix modelling within the HLDA subspace were also examined. Models within this subspace are denoted with the “HLDA+” prefix. The HLDA+EMLLT model is 0.3% worse than the EMLLT model. However, the HLDA+SPAM model gave the best performance of 32.0% WER, which is 1.5% absolute better than the baseline HLDA. This illustrates the importance of compact model representation to yield robust and improved performance.

The final column of Table I depicts the performance of the MPE models. The gain from MPE training is about 3.4–3.8% absolute. The gains from various precision matrix models were retained after MPE training. The WER of the HLDA and HLDA+SPAM MPE models were lowered to 29.8% and 28.5% respectively. This translates to an absolute improvement of 1.3% absolute. Thus, the improvements from structured precision matrices are maintained after MPE training.

As described in Section IV-D, multiple transformations models provide a simple and powerful way of improving modelling accuracies without severely increasing the total number of model parameters. Gaussian clustering is performed in two different ways. For HLDA and STC models, a regression class tree is used to cluster the Gaussian components with an initial speech-silence split. Splitting criterion is based on the Euclidean distance between Gaussian components. This yields the 65-transform (64 speech, 1 silence) HLDA and STC models\(^4\). Gaussian components for the EMLLT models were clustered into 64 groups without an initial speech-silence split and the splitting is based on the Euclidean distance of the vectors of basis coefficients. Table II summarises the WER results for multiple projections HLDA, STC and EMLLT models. These models are 0.8%, 1.0% and 0.6% absolute better than their corresponding single transform models. After 4 MPE iterations, the WER for the HLDA and EMLLT models were both reduced by

\(^4\)The multiple transforms HLDA and STC models were obtained from X. Liu. These models have been trained and decoded using the same setup as described earlier.
3.0% absolute while the STC model achieved a 2.6% absolute WER reduction. After 4 additional MPE iterations, the WER of the 64-transform EMLLT model was 28.3%, 0.9% absolute better than its single-transform model. The slow convergence of the basis update for SPAM models hinders the build of multiple transformation SPAM models. Although it is possible to initialise multiple sets of basis matrices for different cluster of Gaussian components using the method described in Section IV-B, the resulting basis matrices gave a poorer performance than the single transform SPAM models.

B. State-of-the-art Results

So far, the performance of various precision matrix models was presented based on the dev01sub test set for the CTS task. This section compares selected precision matrix models with a CU-HTK LVCSR system [16], [17] similar to that used in the 2003 Rich Transcription (RT03) evaluation\(^5\). The unadapted 28-component HLDA system was chosen as the baseline for comparison. The models were trained on h5etrain03 and evaluated on both dev01sub and eval03. Due to memory constraint, the basis matrices for SPAM models were initialised using the 16-component systems.

The results are shown in Table III. The WERs of the ML HLDA model were 32.3% and 31.7% respectively. The gains from MPE are similar on both test sets, 3.2% and 3.3% respectively. The best single-transform system from before, HLDA+SPAM, was built with 28 Gaussian components per state. Both ML and MPE models consistently outperform the baseline by 1.2% absolute on dev01sub. On eval03, the gains after ML and MPE training were 1.3% and 1.1% absolute, giving the final WER of 27.3% for MPE HLDA+SPAM model. The gains from HLDA+SPAM in Table III were found to be statistically significant\(^6\) at the 95% confidence level. Although the 64-transform 16-component EMLLT model is 0.8% absolute better than the 28-component HLDA model on dev01sub, this gain does not generalise to eval03. Only a 0.3% improvement was obtained on this test set (28.1%).

To examine the performance of MPE trained precision matrices on another task, Broadcast News, a 16-component HLDA+SPAM model was also built to compare with the unadapted HLDA BN HLDA system trained on the bnac+TDT4 (375 hours) date set. These systems were evaluated on the dev03 and eval03 test sets, each consisting of 3 hours data. The results are tabulated in Table IV. The ML base-

<table>
<thead>
<tr>
<th>System</th>
<th>dev01sub</th>
<th>eval03</th>
</tr>
</thead>
<tbody>
<tr>
<td>HLDA</td>
<td>32.3</td>
<td>29.4</td>
</tr>
<tr>
<td>HLDA+SPAM</td>
<td>31.1</td>
<td>27.9</td>
</tr>
</tbody>
</table>

TABLE III
WER PERFORMANCE OF 28-COMPONENT PRECISION MATRIX MODELS ON dev01sub AND eval03 FOR CTS ENGLISH TASK

<table>
<thead>
<tr>
<th>System</th>
<th>dev03</th>
<th>eval03</th>
</tr>
</thead>
<tbody>
<tr>
<td>HLDA</td>
<td>16.3</td>
<td>13.6</td>
</tr>
<tr>
<td>HLDA+SPAM</td>
<td>15.7</td>
<td>13.2</td>
</tr>
</tbody>
</table>

TABLE IV
WER PERFORMANCE OF 16-COMPONENT PRECISION MATRIX MODELS ON dev03 AND eval03 FOR BN ENGLISH TASK

The results are shown in Table III. The WERs of the ML HLDA model were 32.3% and 31.7% respectively. The gains from MPE are similar on both test sets, 3.2% and 3.3% respectively. The best single-transform system from before, HLDA+SPAM, was built with 28 Gaussian components per state. Both ML and MPE models consistently outperform the baseline by 1.2% absolute on dev01sub. On eval03, the gains after ML and MPE training were 1.3% and 1.1% absolute, giving the final WER of 27.3% for MPE HLDA+SPAM model. The gains from HLDA+SPAM in Table III were found to be statistically significant\(^6\) at the 95% confidence level. Although the 64-transform 16-component EMLLT model is 0.8% absolute better than the 28-component HLDA model on dev01sub, this gain does not generalise to eval03. Only a 0.3% improvement was obtained on this test set (28.1%).

To examine the performance of MPE trained precision matrices on another task, Broadcast News, a 16-component HLDA+SPAM model was also built to compare with the unadapted HLDA BN HLDA system trained on the bnac+TDT4 (375 hours) date set. These systems were evaluated on the dev03 and eval03 test sets, each consisting of 3 hours data. The results are tabulated in Table IV. The ML base-

<table>
<thead>
<tr>
<th>System</th>
<th>dev03</th>
<th>eval03</th>
</tr>
</thead>
<tbody>
<tr>
<td>HLDA</td>
<td>16.3</td>
<td>13.6</td>
</tr>
<tr>
<td>HLDA+SPAM</td>
<td>15.7</td>
<td>13.2</td>
</tr>
</tbody>
</table>

VI. CONCLUSIONS

This paper has presented discriminative training of precision matrix models for Large Vocabulary Continuous Speech Recognition (LVCSR) systems. The discriminative criterion selected was MPE as it was found to yield good performance on LVCSR diagonal covariance systems. The structured approximation of precision matrices was described within a generic framework of basis superposition, which subsumes many existing models including the Semi-Tied Covariance (STC), Extended MLLT (EMLLT) and Subspace for Precision and Mean (SPAM) models. These models have efficient likelihood calculation which leads to efficient decoding.

\(^{5}\)See http://htk.eng.cam.ac.uk/docs/cuhtk.shtml

\(^{6}\)Significance tests were carried out using the NIST Scoring Toolkit
Various issues concerning training these models on LVCSR systems were addressed. Computational tractability and memory requirement are two important factors that determine the efficiency of the systems. Issues with high computational cost and slow convergence of the basis matrix update were overcome using good initialization schemes. This also allows the models to be trained by updating only the basis coefficients, which is more efficient and requires significantly less memory. The inefficiency in solving the QEP to find the smoothing constant for the SPAM models was alleviated by using a pseudo transformation matrix to mimic the smoothing constant determination process for STC or EMLLT models.

Experimental results reveal that precision matrix models outperform the standard HLDA diagonal covariance matrix system on the CTS English Task. The best performance was achieved by modelling the precision matrices using the SPAM model within a HLDA subspace. 1.1% and 0.3% absolute WER reductions were obtained on conversational telephone speech and broadcast news tasks respectively over the unadapted HLDA model used in the 2003 Rich Transcription evaluation.

ACKNOWLEDGMENT

The authors would like to thank X. Liu, Cambridge University, for his multiple transforms HLDA and STC models. This work was supported by DARPA grant MDA972-02-1-0013. The paper does not necessarily reflect the position or the policy of the US Government and no official endorsement should be inferred.

REFERENCES

Khe Chai Sim received the B.A. and M.Eng degrees in Electrical and Information Sciences from the University of Cambridge, England in 2001. He then received the M.Phil degree in Computer Speech, Text and Internet Technology from the same university in 2002 before joining the Machine Intelligence laboratory, Cambridge University Engineering Department in the same year as a research student. His research interest is in statistical pattern classification and automatic speech recognition. He also worked on the DARPA funded Effective, Affordable and Reusable Speech-to-text (EARS) project from 2002-2005.

Mark Gales studied for the B.A. in Electrical and Information Sciences at the University of Cambridge from 1985-88. Following graduation he worked as a consultant at Roke Manor Research Ltd. In 1991 he took up a position as a Research Associate in the Speech Vision and Robotics group in the Engineering Department at Cambridge University. In 1995 he completed his doctoral thesis: Model-Based Techniques for Robust Speech Recognition supervised by Professor Steve Young. From 1995-1997 he was a Research Fellow at Emmanuel College Cambridge. He was then a Research Staff Member in the Speech group at the IBM T.J. Watson Research Center until 1999 when he returned to Cambridge University Engineering Department as a University Lecturer. He is currently a Reader in Information Engineering and a Fellow of Emmanuel College. Mark Gales is a member of the IEEE and was a member of the Speech Technical Committee from 2001-2004.