A region-specific feature-space transformation for speaker adaptation and singularity analysis of Jacobian matrix

Shakti P. Rath [1,2]*, Lukáš Burget* [1]*, Martin Karafiát* [1]*, Ondřej Glembek* [1] *and Jan Černocký* [1]

[1]Brno University of Technology, Speech@FIT, Božetěchova 2, Brno, Czech Republic.
[2]Department of Engineering, University of Cambridge, Trumpington Street, Cambridge, UK.
{rath, burget, karafiat, glembek, cernocky}@fit.vutbr.cz

Abstract

In this paper, we present an in-depth analysis of a recently proposed method for speaker adaptation. The method involves a region-specific feature-space transformation, which we refer to as $soft$ R-FMLLR. We argue that the method has certain difficulties, the most significant being the fact that it is non-invertible. An analysis that pertains to the singularity of the Jacobian matrix is presented, from which we note that the matrix becomes near-singular at certain points in the feature space. It indicates that the transformation is non-invertible. We observe that under this case maximum likelihood estimation adversely affects the speech recognition performance. Moreover, sufficient statistics do not exist that makes the estimation procedure computationally very expensive. The concerns outlined above render the method to be unattractive. We propose a simple yet important modification, $hard$ R-FMLLR, and show that the associated Jacobian matrix is assured to be full-rank, and it is computationally efficient. On a large vocabulary continuous speech recognition task the performance of the proposed method is shown to be better than $soft$ R-FMLLR. Further, it is comparable to the widely used CMLLR with regression classes, especially when higher number of transforms are used.

1. Introduction

It is well known that the effect of inter-speaker variability on speech originating from different acoustic classes can be significantly different [1]. It is therefore appropriate to modify the widely used model-space transformation technique, constrained maximum likelihood linear regression (CMLLR), to be specific to each acoustic class rather than being global [2]. In [3], it is achieved by dividing the acoustic space into a small number of *homogeneous regions* and allowing each region to have a separate linear transform. The collections of Gaussian mixtures in the HMM (known as regression classes) are used to represent the homogeneous regions. Speaker adaptation is achieved by

$$\hat{\boldsymbol{\mu}}_{jm} = \mathbf{B}_r\boldsymbol{\mu}_{jm} - \mathbf{b}_r\,,\ \hat{\boldsymbol{\Sigma}}_{jm} = \mathbf{B}_r^T\boldsymbol{\Sigma}_{jm}\mathbf{B}_r, \qquad (1)$$

where $(\mathbf{B}_r, \mathbf{b}_r)$ is the CMLLR transform shared with regression class r, and $\boldsymbol{\mu}_{jm}$ and $\boldsymbol{\Sigma}_{jm}$ denote the mean and co-variance of the Gaussian mixtures in the speaker independent (SI) HMM, respectively. We have suggested a similar scheme using linearized vocal tract length normalization [4].

Motivated by [5], recently Kozat et al. [6] presented an alternative method to enable class-specific speaker adaptation. Unlike CMLLR, it is a fully feature-space transformation that "adapts features" to speakers before passing them to the acoustic model [1]. In [6] the *homogeneous regions* are represented by a separate GMM that is used to cluster the feature space. If C denotes the number of mixtures in the GMM the transformation can be formulated as :

$$\mathbf{f} : \mathbb{R}^d \rightarrow \mathbb{R}^d,\ \ \mathbf{y}_t = \mathbf{f}(\mathbf{x}_t) = \sum_{l=1}^{C} \gamma_l^g(\mathbf{x}_t)\mathbf{W}_l\mathbf{x}_t^+, \qquad (2)$$

where $\mathbf{W}_l$ denotes the transform associated with mixtures l; it has a linear part and a bias, i.e., $\mathbf{W}_l = [\mathbf{A}_l\ ;\ \mathbf{b}_l]$. $\mathbf{x}_t$ and $\mathbf{y}_t$ are the speaker un-normalized and the speaker normalized feature vectors at time t, respectively, and d is the dimension of the vectors. $\mathbf{x}_t^+$ is the extended feature vector defined as $\mathbf{x}_t^{+T} = [\mathbf{x}_t^T\ ;\ 1]^T$. $\gamma_l^g(\mathbf{x}_t)$ denotes the posterior probability of mixture l computed w.r.t. $\mathbf{x}_t$. The transforms are estimated using the maximum likelihood (ML). We will refer to this scheme as $soft$ region-specific FMLLR ($soft$ R-FMLLR) in this paper. The same feature-space transformation was briefly discussed in [10].

One of the objectives of this paper is to present a detailed study of the $soft$ R-FMLLR model and discuss the underlying difficulties. Further, we present a simple yet important extension that overcomes the difficulties.

The main concern with Eq. 2 is that since it involves a linear combination of two or more linear transforms, it *may not be an invertible function*. In such case, the Jacobian matrix, $\frac{d\mathbf{y}_t}{d\mathbf{x}_t}$, would become singular that will force the likelihood function to go to $-\infty$. This would degenerate the system and hence might lead to a degraded speech recognition performance. Although the authors discussed the problem briefly in [6], steps had not been taken to address it. The observations made by the authors are as follows: (i) When the transforms were initialized with the *identity matrix* for (iterative) ML estimation, the benefit expected due to region-specific transformation was not observed. In fact, the recognition performance was worse than the performance given by global transformation. (ii) In contrast, when the transforms were initialized with the *global* transform, the expected improvement was indeed achieved.

In this paper, we present a detailed study of the two cases discussed above. The observations made by us are as follows: (i) In the first case (initialization with the identity matrix), the Jacobian matrix of the model in fact turned out to be *very close to be singular* over a small number of feature vectors in the test set. This is the critical problem with the technique and the

S. P. Rath was supported by Detonation project within SoMoPro. The work was also supported by Technology Agency of the Czech Republic grant No. TA01011328.

[1]Such feature processing is useful with deep neural network based acoustic models [7, 8], which has recently been proved to be much more effective than the traditional GMM-based systems [9]. Although CMLLR with multiple transforms can be equivalently expressed as a feature-space operation, it still depends on the acoustic model through the regression-class definition to retrieve the transform-to-class mapping. Hence CMLLR is not a fully feature-space operation.

reason for the degradation in the performance. (ii) In the second case, the number of feature vectors where the Jacobian matrix is near singular was found to be extremely small. As a result, the detrimental effect was much less significant.

Such observations were made when we conducted an in-depth analysis pertaining to the singularity of the Jacobian matrix (Section 2.2). We conclude from the study that the transformation is highly likely to be non-invertible and in such case ML estimation adversely affects the recognition performance. In addition, we noted that even with global initialization, the performance was poorer than CMLLR. Moreover, sufficient statistics of finite dimension do not exist; as a consequence, repeated processing of adaptation data would be required while (iteratively) estimating the transforms. Hence the computational cost becomes very high. The difficulties described above render $soft$ R-FMLLR to be unattractive.

We present a simple yet important extension – we propose to quantize the original mixture posteriors such that each feature vector is transformed by only one region-specific transform. The resulting transformation is termed as $hard$ R-FMLLR. It offers the following advantages over $soft$ R-FMLLR:

1. The Jacobian matrix is ensured to be full-rank over all points in the feature space irrespective of the initialization.
2. The recognition performance is shown to be better than $soft$ R-FMLLR on an LVCSR task.
3. It allows collection of sufficient statistics; hence, it is computationally more efficient (faster) than $soft$ R-FMLLR.

Further, the performance of $hard$ R-FMLLR is shown to be comparable to CMLLR, especially when higher number (4 or more) of transforms are used.

The organization of the rest of the paper is as follows. In Section 2, we discuss $soft$ R-FMLLR [6] and the associated difficulties. The proposed $hard$ R-FMLLR is described in Section 3. The experimental results are presented in Section 4. Finally, we conclude in Section 5.

2. Soft R-FMLLR

Our implementation of $soft$ R-FMLLR differs from [6] in that we use the Expectation Maximization (EM) algorithm for ML estimation of the transforms, whereas in the previous work the likelihood function was directly maximized. The GMM is created from a well trained SI HMM model by iteratively merging the closest Gaussian mixtures [11] until the required number of components is obtained. The EM auxiliary function for $soft$ R-FMLLR can be shown to be[2]

$$\mathcal{Q}(\{\mathbf{W}_l\}_{l=1}^{C}) = \sum_{t=1}^{T} \log|\mathbf{J}_{\mathbf{x}_t}| + \sum_{l=1}^{C}\left\{ \mathrm{tr}\left(\mathbf{W}_l \mathbf{K}_l^{\,T}\right) - \frac{1}{2}\sum_{p=1}^{C} \mathrm{vec}\left(\mathbf{W}_l\right)^T \mathbf{G}_{lp} \mathrm{vec}\left(\mathbf{W}_p\right) \right\} \quad (3)$$

Note that the auxiliary function (and the likelihood function) is valid only for invertible transformation. The first and second order statistics, $\mathbf{K}_l$ and $\mathbf{G}_{lp}$, are as follows:

$$\mathbf{K}_l = \sum_{t=1}^{T}\sum_{jm} \gamma_l^g(\mathbf{x}_t)\gamma_{jm}(t)\boldsymbol{\Sigma}_{jm}^{-1}\boldsymbol{\mu}_{jm}(\mathbf{x}_t^+)^T, \quad (4)$$

$$\mathbf{G}_{lp} = \sum_{t=1}^{T}\sum_{jm} \gamma_l^g(\mathbf{x}_t)\gamma_p^g(\mathbf{x}_t)\gamma_{jm}(t)\ \mathbf{x}_t^+(\mathbf{x}_t^+)^T \otimes \boldsymbol{\Sigma}_{jm}^{-1}, \quad (5)$$

[2]The steps for derivation of auxiliary function are similar to our earlier work in [12]. The expressions are also valid for full co-variances.

where $\otimes$ denotes the Kronecker product. $\boldsymbol{\mu}_{jm}$ and $\boldsymbol{\Sigma}_{jm}$ are the mean vector and the co-variance matrix of mixture g_{jm} in the HMM set, respectively. vec$(\cdot)$ is the column-wise vectorization operation on the matrix and tr$(\cdot)$ is the matrix trace operation. The number of (existing) statistics to be collected is $C + C^2$. The Jacobian matrix, $\mathbf{J}_{\mathbf{x}_t}$, appearing in Eq. 3 is

$$\mathbf{J}_{\mathbf{x}_t} = \frac{d\mathbf{y}_t}{d\mathbf{x}_t} = \sum_{l=1}^{C} \mathbf{A}_l \left(\mathbf{x}_t \left(\frac{d\gamma_l^g(\mathbf{x}_t)}{d\mathbf{x}_t} \right)^T + \gamma_l^g(\mathbf{x}_t)\mathbf{I} \right), \quad (6)$$

$$\frac{d\gamma_l^g(\mathbf{x}_t)}{d\mathbf{x}_t} = \gamma_l^g(\mathbf{x}_t)\left(\boldsymbol{\Sigma}_l^{g\,-1}\left(\boldsymbol{\mu}_l^g - \mathbf{x}_t\right) - \sum_{p=1}^{C} \gamma_p^g(\mathbf{x}_t)\boldsymbol{\Sigma}_p^{g\,-1}\left(\boldsymbol{\mu}_p^g - \mathbf{x}_t\right) \right).$$

We note that the Jacobian matrix is a *highly non-linear* function of the feature vector, $\mathbf{x}_t$. A closed-form solution that maximizes the auxiliary function, $\mathcal{Q}$, w.r.t. $\{\mathbf{W}_l\}$ does not exist. The Broyden-Fletcher-Goldfarb-Shanno (BFGS) [13] is used for the optimization – 100 iterations are used in our experiments.

2.1. Limitations

As discussed earlier, since Eq. 2 involves a linear combination of two or more ($C \geq 2$) transforms, it may not be an invertible function, i.e., $\mathbf{y}_t = \mathbf{f}(\mathbf{x}_t)$ may be a *many*-to-*one* mapping, i.e.,

$$card\left\{ \mathbf{x}_t \ \mid \ \mathbf{y}_t = \mathbf{f}(\mathbf{x}_t)\ ,\ \mathbf{x}_t \in \mathbb{R}^d \right\} \geq 1. \quad (7)$$

In other words, a certain point in the range of the function may correspond to more than one points in the domain. From the practical point of view, the consequence of non-invertibility is as follows: it would force the Jacobian matrix to become singular at certain points in the feature space[3] and hence as we note from Eq. 3 the auxiliary function (and the log-likelihood function) would go to $-\infty$. In Section 2.2, we present a test conducted to examine the singularity of Jacobian matrix and study its implications on the recognition performance.

Moreover, we note from Eq. 3 through 6 that, although sufficient statistics exist for the second and third terms of the auxiliary function, they do not exist for the term involving the Jacobian matrix. Hence, it is necessary to process the data as many times as the number of optimization iterations used while optimizing the transforms; this is computationally very expensive (100 iterations are used both in our implementation and in [6]).

2.2. Proposed test for singularity of Jacobian matrix

We conducted the following experiments with the aim to detect the feature vectors where the Jacobian matrix is singular. Refer to Section 4 for the experimental set-up.

(1) In the first experiment, we used the transforms of the final iteration of ML estimation and computed the ratio of the largest to smallest singular values (singular value ratio, SVR) of the Jacobian matrix at all feature vectors in the test set:

$$\eta(\mathbf{x}_t) = \frac{\lambda_{max}\left(\mathbf{J}_{\mathbf{x}_t}\right)}{\lambda_{min}\left(\mathbf{J}_{\mathbf{x}_t}\right)}, \quad (8)$$

where $\mathbf{J}_{\mathbf{x}_t}$ is the Jacobian matrix evaluated at $\mathbf{x}_t$ (Eq. 6). A large $\eta(\mathbf{x}_t)$ therefore would imply that $\mathbf{J}_{\mathbf{x}_t}$ is close to be singular[4] at $\mathbf{x}_t$. The following criterion is used to compute the average number of $unsafe$ frames, i.e., the points where $\eta(\mathbf{x}_t) \geq 10^5$:

$$N_u = \frac{1}{R}\sum_{r=1}^{R}\sum_{t=1}^{T_r} \delta(\mathbf{x}_t)\ ,\ \delta(\mathbf{x}_t) = \begin{cases} 1 & \text{if } \eta(\mathbf{x}_t) \geq 10^5 \\ 0 & \text{otherwise} \end{cases}$$

[3]This is based on the inverse function theorem, Chapter 9 in [14].

[4]Although the test set includes a large number of feature vectors, the vectors where the Jacobian matrix is exactly singular may not appear. Hence, SVR has been used as a measure for closeness to singularity.

Table 1: Jacobian test on $soft$ R-FMLLR: Average auxiliary function values ($\mathcal{Q}$) and average number of $unsafe$ frames (N_u) are shown for *with* and *without* feature compensation and different initialization conditions. Number of frames per-speaker is about 11090.

# of transforms per speaker	initialization with: Identity matrix (case I)			initialization with: Global transform (case G)			initialization with: Diagonal transforms (case D)		
	feature compensation			feature compensation			feature compensation		
	without		with	without		with	without		with
C	N_u	$\mathcal{Q}$	$\mathcal{Q}$	N_u	$\mathcal{Q}$	$\mathcal{Q}$	N_u	$\mathcal{Q}$	$\mathcal{Q}$
No Adaptation	-	-19.54	-19.54	-	-19.54	-19.54	-	-19.54	-19.54
1 (global)	0	-18.43	-18.43	0	-18.03	-18.03	0	-18.23	-18.23
2	4.05	-18.69	-17.91	0.37	-17.53	-17.31	9.61	$-\infty$	-17.76
4	2.60	-17.67	-17.26	0.22	-16.95	-16.78	5.1	-17.79	-17.13
6	0.71	-17.06	-16.87	0.10	-16.48	-16.36	1.99	-16.85	-16.48

Table 2: Jacobian test on $soft$ R-FMLLR: WER (%) *with* and *without* feature compensation. Column 1 indicates number of transforms per speaker. NA=No adaptation.

C	initialization with					
	Identity (case I) compensation		Global (case G) compensation		Diagonal (case D) compensation	
	without	with	without	with	without	with
NA	43.0	43.0	43.0	43.0	43.0	43.0
1	39.9	39.9	39.8	39.8	39.8	39.8
2	41.2	40.2	39.7	39.5	42.0	40.2
4	40.6	39.6	39.2	39.0	41.6	40.3
6	40.4	39.3	39.0	38.9	40.6	39.5

where R is the number of speakers in the test set and T_r is the number of frames from speaker r. Three cases are considered – the region-specific transforms being initialized with identity matrix (case I), the estimated global transform (case G) and the diagonal transforms (case D). The average (per-speaker, per-frame) auxiliary function value, $\mathcal{Q}$, and N_u, are presented in Table 1 (shown as *without* feature compensation).

(2) In another experiment, the detected unsafe frames were *removed* from the test set in all iterations of estimation. The value of auxiliary function, $\mathcal{Q}$, of the final iteration is presented in the same Table (shown as *with* feature compensation).

The following observations pertain to the cases when two or more transforms are used: (i) The number of unsafe frames (N_u) is observed to be non-zero in all initializing conditions (I, G, D). It constitutes a small fraction of the total number of frames ($\approx$ 11090 frames per-speaker). (ii) Under cases I and D and *with* compensation, the value of auxiliary function is significantly higher than *without* compensation. (iii) In particular, in case D with two transforms being used, the Jacobian matrix turned out to be exactly singular. This forced the auxiliary function to reach $-\infty$. *With* compensation, the auxiliary function increased from $-\infty$ to -17.76. (iv) In case G, although the number of unsafe frames is very small, it is still non-zero.

2.3. Results of recognition experiments

Percentage of word error rates (WER) comparing with and without feature compensation are presented in Table 2. *With* compensation, the unsafe frames were again removed during decoding. The following observations can be made:

(i) Under cases I and D, *without* compensation, the WER with region-specific transformation is poorer than that with global transformation. Significant reduction in WER results *with* compensation. (ii) Under these cases (I and D) and *with* compensation, the WER with region-specific transformation is inconsistent compared to global transformation. This is because the Jacobian based feature compensation is not meant to provide complete removal of all unsafe frames – it can only *reduce* them based on the threshold set on η. (iii) In case G, N_u is close to zero. In this case, WER with region-specific transformation can be observed to be better than global transformation *without* compensation. However, relatively small additional improvement can still be seen *with* compensation.

From the experiments, we note that there are certain feature vectors in the acoustic space where the Jacobian matrix is singular or is near-singular. Therefore, the transformation is most likely to be non-invertible. Further, the transforms obtained by ML estimation adversely affect the recognition performance. Although initialization with the global transform leads to a very small N_u, it is based on heuristics, hence it may not be reliable.

3. Proposed Hard R-FMLLR

We propose to quantize the original GMM posteriors such that they take binary values (i.e., 0 or 1). In this section we show that the resulting feature transformation model is free of the difficulties associated with the previous model.

With the quantized posteriors only one region-specific transform would normalize each feature vector over the complete space, instead of a linear combination. To formulate the transformation model with $hard$ R-FMLLR, let us define the indicator function as follows:

$$1_{(\mathcal{R}_l)}(\mathbf{x}_t) = \begin{cases} 1, & \mathbf{x}_t \in \mathcal{R}_l, \\ 0, & \text{otherwise} \end{cases}, \tag{9}$$

where $\mathcal{R}_l$ is the collection of feature vectors (or, the region in the acoustic space) where the quantized posterior of mixture l in the GMM is unity. Hence, the space of feature vectors is divided into C *disjoint regions* by the GMM. The speaker normalization model can now be expressed as

$$\mathbf{y}_t = \mathbf{f}(\mathbf{x}_t) = \sum_{l=1}^{C} 1_{(\mathcal{R}_l)}(\mathbf{x}_t)\, \mathbf{W}_l\, \mathbf{x}_t^+. \tag{10}$$

The EM auxiliary function for $hard$ R-FMLLR can be obtained from that of $soft$ R-FMLLR using the fact that the posterior probabilities are binary:

$$\mathcal{Q} = \sum_{t=1}^{T} \log|\mathbf{J}_{\mathbf{x}_t}| + \sum_{l=1}^{C} \text{tr}\left(\mathbf{W}_l \mathbf{K}_l{}^T\right) - \frac{1}{2}\text{vec}\left(\mathbf{W}_l\right)^T \mathbf{G}_l \text{vec}\left(\mathbf{W}_l\right),$$

where the second order statistic is

$$\mathbf{G}_l = \sum_{\mathbf{x}_t \in R_l} \sum_{jm} \gamma_{jm}(t)\, \mathbf{x}_t^+ (\mathbf{x}_t^+)^T \otimes \mathbf{\Sigma}_{jm}^{-1} \tag{11}$$

(the first order statistic, $\mathbf{K}_l$, can be similarly obtained from Eq. 4) and the Jacobian matrix is

$$\mathbf{J}_{\mathbf{x}_t} = \frac{d\mathbf{y}_t}{d\mathbf{x}_t} = \sum_{l=1}^{C} \mathbf{A}_l \left(\mathbf{x}_t \left(\frac{d\, 1_{(R_l)}(\mathbf{x}_t)}{d\mathbf{x}_t} \right)^T + 1_{(R_l)}(\mathbf{x}_t)\mathbf{I} \right). \tag{12}$$

Table 3: WER (%) with CMLLR and R-FMLLR. The initialization conditions are shown in bracket. Results with $soft$ R-FMLLR correspond to the case when feature compensation was not used. Column 1 indicates number of transforms per speaker.

# of transforms per speaker C	CMLLR (diagonal)	R-FMLLR $soft$ (global)	R-FMLLR $hard$ (diagonal)
No adaptation	43.0	43.0	43.0
1 (global)	39.8	39.8	39.8
2	39.0	39.7	39.4
4	38.8	39.2	38.9
6	**38.8**	39.0	**38.8**
8	39.0	39.1	39.0

Using the Dirac delta to compute the derivative of the indicator function, the Jacobian matrix turns out to be

$$\mathbf{J}_{\mathbf{x}_t} = \sum_{l=1}^{C} 1_{(R_l)}(\mathbf{x}_t)\mathbf{A}_l. \tag{13}$$

Unlike the Jacobian matrix of $soft$ R-FMLLR, Eq. 13 has a simple structure – it depends on the feature vector via the indicator function, which implies that the Jacobian matrix is equal to one of the C transforms at each point in the feature space. Hence, it is assured to be non-singular[5] over the entire space.

Further, using the disjoint property of the regions, the Jacobian determinant simplifies to

$$\sum_{t=1}^{T} \log|\mathbf{J}_{\mathbf{x}_t}| = \sum_{l=1}^{C} \beta_l \log|\mathbf{A}_l|, \tag{14}$$

where β_l is the number of frames in the adaptation set belonging to region $\mathcal{R}_l$. β_l is the order zero statistic. Therefore, with quantized posterior, sufficient statistics of finite dimension exist, which are $\{\beta_l, \mathbf{K}_l, \mathbf{G}_l\}_{l=1}^{C}$. They can be collected by processing adaptation data *only once*. The total number of statistics to be collected with $hard$ R-FMLLR is $3C$. In contrast $soft$ R-FMLLR requires as many times processing of data as the number of optimization iterations. Hence, $hard$ R-FMLLR is computationally more efficient than its $soft$ counterpart.

Further, since the region-specific transforms independently contribute to the auxiliary function, they can be optimized separately. The gradient ascent algorithm is used to iteratively optimize the transforms. 100 iterations are performed after initializing them with the diagonal matrices.

Another difference between $soft$ and $hard$ R-FMLLR is that with $soft$ R-FMLLR the number of statistics to be collected is $C + C^2$ (each is a matrix), whereas with $hard$ R-FMLLR, it is only C (scalars) $+ 2C$ (matrices).

3.1. Comparative analysis and future direction

The comparison between CMLLR and both form of R-FMLLR is summarized in Table 4. We emphasize that R-FMLLR is favorable for deep neural network (DNN), which has recently been shown to be more effective than GMM based systems. In this "hybrid" frame-work, an artificial neural network is trained to output HMM state level posterior probabilities [15, 9]. Since DNNs do not contain Gaussian mixtures, the regression classes are not defined; hence CMLLR with multiple transforms cannot be incorporated into this frame-work. On the other hand, since R-FMLLR is a fully feature-space transformation we can generate the speaker adapted features and pass them as the input to the neural network, following ideas similar to [7, 8]. Therefore, R-FMLLR has an advantage over CMLLR in this sense.

[5] We assume each of the C transforms to be full-rank.

Table 4: A comparative analysis of CMLLR and R-FMLLR.

Attributes	CMLLR (regression)	R-FMLLR $soft$	R-FMLLR $hard$
Jacobian Matrix	Full Rank	Singular	Full Rank
Sufficient Stats (Fast?)	Yes (Fast)	No (Slow)	Yes (Fast)
Memory Requirement to store existing statistics	$O(C)$	$O(C^2)$ $(*)$	$O(C)$
Favors Neural Network?	No	Yes	Yes

$(*)$ The order-zero statistic of $soft$ R-FMLLR does not exist.

Our future direction includes investigation of R-FMLLR with the DNN systems. We are interested mainly in the proposed hard version because of the advantages it offers.

4. Experimental setup and Results

For the experiments [12], the SI HMM model was trained on the *ctstrain-04* training set, which is a subset of the *h5train-03* set. The training set contains about 278 hours of speech from Switchboard I, II and Call Home English. Test was done on the *Hub5 Eval-01* test set, which was used during NIST 2001 CTS evaluation. It consists of 3 subsets of 20 conversations from Switchboard-1, Switchboard-2 and Switchboard cellular corpora and contains more than 6 hours of speech. Bi-gram language model from AMI speech recognition system for NIST Rich Transcriptions 2007 was used during decoding [16]. 39-dimensional MFCC features that consist of 13 (C_1 to C_{12} and C_0) static, Δ and $\Delta\Delta$ components were used. Speaker-wise cepstral mean and cepstral variance normalization were performed both during training and test. 3-state cross-word triphone HMM models with 20 (diagonal co-variances) mixtures per state were used. The un-supervised adaptation was used and speaker adaptive training [2] was not applied. The test set included data from 120 speakers. The duration of test data per speaker was 3 minutes in average.

The WER (%) comparing the three methods are presented in Table 3. The following observations can be made. (1) The WER with all three methods decreased monotonically from a single transform up to 6 transforms per speaker, where the best performance is achieved. (2) The WER with $hard$ R-FMLLR and CMLLR are comparable in all cases, except with 2 transforms where $hard$ R-FMLLR is slightly inferior. (3) The best performance with $hard$ R-FMLLR is same as the best with CMLLR (with 6 transforms). (4) The WER with $soft$ R-FMLLR is consistently inferior to both CMLLR and $hard$ R-FMLLR. We may expect the soft classification of the frames into regions (as with $soft$ R-FMLLR) to provide better performance than hard classification. We believe, such effect has been diminished by the adverse effect caused by the fact that the Jacobian matrix is singular. (5) With 6 transforms, the WER with $soft$ R-FMLLR is slightly inferior to the other two. However, as mentioned before, since sufficient statistics of $soft$ R-FMLLR do not exist, the *computational cost becomes very high*, which makes the method prohibitive to use in practice.

5. Conclusions

In this paper, the difficulties associated with $soft$ R-FMLLR are addressed. By analyzing the Jacobian matrix, it was concluded that the transformation is most likely to be non-invertible and in this case ML estimation adversely affects the performance. A new transformation, $hard$ R-FMLLR, is presented. It is shown that the performance of the proposed method is better than $soft$ R-FMLLR and it is computationally more efficient.

6. References

[1] G. Fant, "Non-uniform vowel normalization," *STL-QPSR*, vol. 16, no. 2-3, pp. 1–19, 1975.

[2] M. J. F. Gales, "Maximum Likelihood Linear Transformations for HMM-Based Speech Recognition," *Computer Speech and Language*, vol. 12, no. 2, pp. 75–98, 1998.

[3] M. J. F. Gales, "The generation and use of regresion class trees for MLLR adaptation," in *techical report CUED/F-INFENG/TR263*, Cambridge University, Cambridge, U. K., 1996.

[4] S. P. Rath and S. Umesh, "Acoustic class specific VTLN-warping using regression class trees," in *Proc. Interspeech*, Brighton, September 2009.

[5] D. Povey, B. Kingsbury, et al., "fMPE: Discriminatively trained features for speech recognition," in *Proc. of IEEE ICASSP*, 2005.

[6] S. S. Kozat, K. Visweswariah, and R. Gopinath, "Feature adaptation based on Gaussian posteriors," in *Proc. of IEEE ICASSP*, Toulouse, 2006.

[7] S. P. Rath, D. Povey, K. Vesely, and J. Cernocky, "Improved feature processing for Deep Neural Networks," in *Proc. of Interspeech*, 2013.

[8] F. Seide, G. Li, X. Chen, and D. Yu, "Feature engineering in context-dependent Deep Neural Networks for conversational speech transcription," in *Proc. of IEEE ASRU*, Dec. 2011, pp. 24–29.

[9] G. Hinton, L. Deng, et al., "Deep Neural Networks for acoustic modeling in speech recognition: The shared views of four research groups," *Signal Processing Magazine, IEEE*, vol. 29, no. 6, pp. 82 –97, Nov. 2012.

[10] H. Liao and M. J. F. Gales, "Joint uncertainty decoding for noise robust speech recognition," in *Proc. of Interspeech*, 2005.

[11] D. Povey, L. Burget, et al., "The subspace Gaussian mixture model – A structured model for speech recognition," *Comput. Speech Lang.*, vol. 25, pp. 404–439, April 2011.

[12] S. P. Rath, M. Karafiat, O. Glembek, and J. Cernocky, "A factorized representation of FMLLR transform based on QR-decomposition," in *Proc. of Interspeech*, 2012.

[13] J. Nocedal and S. J. Wright, *Numerical Optimization*, Springer, Aug 1999.

[14] W. Rudin, *Principles of Mathematical Analysis*, McGraw-Hill, New York, 3d ed., edition, 1976.

[15] H. A. Bourlard and N. Morgan, *Connectionist Speech Recognition: A Hybrid Approach*, Kluwer Academic Publishers, Norwell, MA, USA, 1993.

[16] T. Hain, L. Burget, et al., "The AMIDA 2009 meeting transcription system," in *Proc. of Interspeech*, China, 2010.