Extended Partial Distance Elimination and Dynamic Gaussian Selection for Fast Likelihood Computation

Ghazi Bouselmi1, Jun Cai1,2
1Speech Group, LORIA-CNRS & INRIA
BP 239, 54600 Vandoeuvre-les-Nancy, France
2Dept. of Cognitive Science, Xiamen Univ., 361005 Xiamen, China
bouselm@loria.fr, Jun.Cai@loria.fr

Abstract
A new fast likelihood computation approach is proposed for HMM-based continuous speech recognition. This approach is an extension of the partial distance elimination (PDE) technique. Like PDE, the extended PDE (EPDE) approach aims at finding the most prominent Gaussian in a GMM for a given observation, and approximating the GMM’s likelihood with the identified Gaussian. EPDE relies on a novel selection criterion in order to achieve greater time efficiency at the cost of slight degradation of recognition accuracy. This novel criterion has been combined with a dynamic Gaussian selection technique for greater recognition accuracy. Tests on TIMIT corpora shows a satisfying computation time saving of 7.3% at the same error level as PDE. Compared to a baseline, the methods we propose have also achieved a significant reduction in the number of computations of 71.5% at the same error level as PDE.

Index Terms: fast likelihood computation, partial distance elimination, Gaussian selection, hidden Markov models, speech recognition.

1. Introduction
Nowadays, most large vocabulary continuous speech recognition (LVCSR) systems rely on continuous density HMMs (CD-HMMs) for acoustic modelling of speech signals. In such systems, the triphone HMM acoustic models typically comprise 2000 to 6000 model states, each of which is described as a Gaussian mixture model (GMM) with 8 to 64 multi-dimensional Gaussians. Even more Gaussians can be used in each GMM in order to enhance the recognition accuracy. The large number of acoustic model parameters implies an intensive computational load and a long computation time in the process of likelihood computation for each observation. As stated in [1], the state likelihood estimation takes from 30% to 70% of the total recognition time, depending on the task in hands and the complexity of the models. Along with language model searching, likelihood computation is so time consuming that most of LVCSR systems run at several times slower than real time.

To speedup the likelihood computation, we propose a new method called “Extended PDE”. It uses a novel partial elimination criterion to achieve further time saving than PDE [2, 3]. We also propose a combination of this new criterion with a dynamic Gaussian selection (DGS) technique [4] to minimize the approximation error.

The article is organized as follows. In section 2, two approaches for fast likelihood computation are reviewed, namely the nearest neighbour approximation (PDE) and VQ-based Gaussian selection. In section 3, we present our new PDE criterion, which is called “Extended PDE”, for fast likelihood computation. The combination of this new criterion with a dynamic Gaussian selection technique is presented in section 4. Section 5 reports a performance analysis of the new methods, along with recognition experiments on TIMIT corpus. Finally, the article ends in a discussion and a brief conclusion.

2. Background
Originally, Bei et al. [2] developed the “partial distance elimination” technique (PDE) to speed up vector quantisation (VQ) processing by eliminating unnecessary computations. In such an issue, the goal is to classify a vector \( x \), of dimension \( N \), into the codeword \( \tilde{c} \) according to Equ. (1). In the case of Euclidean distance, Equ. (1) can be expressed as Equ. (2).

\[
\tilde{c} = \arg\min_{c \in C} \{ D(x,c) \} \tag{1}
\]

\[
\tilde{c} = \arg\min_{c \in C} \{ \sum_{j=1}^{N} (x_j - c_j)^2 \} \tag{2}
\]

where \( C = \{ c_1, ..., c_N \} \) is the set of codewords, \( D \) a distortion measure and \( x_j \) (resp. \( c_j \)) is the \( j \)th element of the vector \( x \) (resp. \( c \)).

If we define the partial distortion at the rank \( k \), between a vector \( x \) and a codeword \( c \) as in Equ. 3, we would have the relations of Equ. (4) and (5).

\[
D_k(x, c) = \sum_{j=1}^{k} (x_j - c_j)^2 \tag{3}
\]

\[
D_k(x, c) = D_{k-1}(x, c) + (x_k - c_k)^2, \forall 1 < k \leq N \tag{4}
\]

\[
D(x, c) = D_N(x, c) \tag{5}
\]

One can easily see that the partial distortion is monotonically increasing over the dimensionality of the vectors. The main idea of the PDE approach is to utilise the monotonicity of the distortion used in the VQ. Knowing the value of the minimal distortion \( \bar{D} \) between a vector \( x \) and the codewords \( \{ c_1, ..., c_N \} \), a part of the computation of the distortions between \( x \) and the rest of the codewords \( \{ c_{k+1}, ..., c_N \} \) can be truncated. Indeed, for a sub-optimal codeword \( \tilde{c} \in \{ c_{k+1}, ..., c_N \} \) (satisfying \( D(x, \tilde{c}) > \bar{D} \)), there exists a rank \( \tilde{k} \) where \( D_{\tilde{k}}(x, \tilde{c}) > \bar{D} \). When the previous condition is met at a rank \( \tilde{k} \), one can predict that \( D(x, \tilde{c}) > \bar{D} \) and that \( \tilde{c} \) is not the closest codeword to \( x \). Thus, the computation of partial distortions \( D_{\tilde{k}+1}(x, \tilde{c}) ... D_N(x, \tilde{c}) \) can be discarded. A computational time reduction is achieved without any approximation error for the minimal distortion or the classification of the vector \( x \).
2.1. PDE in likelihood computation

For CD-HMMs continuous speech recognition systems, the probability density of each HMM state \( G \) is modeled as a GMM, as in Eq. (6). In this equation, \( d \) is the number of Gaussians in the GMM, \( N(\mu_i, \Sigma_i) \) are Gaussian distributions and \( \omega_i \) are weights with \( \sum_i \omega_i = 1 \). The emission probability of an observation \( x \) by an HMM state \( G \) can be expressed as in Eq. (7), in which \( p(x|N(\mu_i, \Sigma_i)) \) is the emission probability for observation \( x \) by the Gaussian \( N(\mu_i, \Sigma_i) \), and is expressed as in Eq. (8). For diagonal variance matrices \( \Sigma_i \), the likelihood of emission of \( x \) by a Gaussian \( N(\mu_i, \Sigma_i) \) can be expressed as in Eq. (9). It can be seen that, in Eq. (9), the emission likelihood of a Gaussian is a weighted distortion measure between \( x \) and the mean of the Gaussian, where the weight is the variance of the Gaussian.

\[
p(x|G) = \sum_{i=1}^{d} \omega_i N(\mu_i, \Sigma_i) \tag{6}
\]
\[
p(x|\mu_i, \Sigma_i) = \frac{1}{(2\pi)^{d/2} |\Sigma_i|^{1/2}} e^{-\frac{1}{2}(x-\mu_i)^T \Sigma_i^{-1} (x-\mu_i)} \tag{7}
\]
\[
\ell = \log p(x|N(\mu_i, \Sigma_i)) = Z_i - \log(2\pi)^{d/2} |\Sigma_i|^{1/2} - \frac{1}{2} \sum_{k=1}^{d} (x_k - \mu_{ik})^2 \tag{8}
\]

where \( Z_i = \log(\frac{1}{(2\pi)^{d/2} |\Sigma_i|^{1/2}}) \) is a constant for the Gaussian \( i \) and \( x \) (resp. \( \mu_i, \Sigma_i \)) is the \( k \)th element of \( x \) (resp. \( \mu_i, \Sigma_i \)).

Besides, for an observation \( x \), only few Gaussians have a prominent contribution in the emission probability of a GMM. Our experiments, sketched in Table 1, show that the closest Gaussian to an observation \( x \) contributes by an average of 85.9% of the total probability of the whole GMM. This understanding led to considering the approximation of the emission likelihood of a GMM with the likelihood of the closest Gaussian to the observation under consideration. The likelihood of emission of an observation \( x \) by a GMM \( G \) would be expressed as in Eq. (10).

\[
\ell(x|G) \approx \max_{i=1}^{d} \left\{ \log(\omega_i) + Z_i - \frac{1}{2} \sum_{k=1}^{d} (x_k - \mu_{ik})^2 / \Sigma_{ik} \right\} \tag{10}
\]

In the light of such an approximation, the likelihood computation for a GMM becomes a VQ problem where the code-words are the Gaussians and the distortion measure is the likelihood of each Gaussian. Thus, likelihood computation (VQ problem as defined in section 2) can benefit from the PDE approach to reduce time consumption. For that matter, Pellom et al. [3] have used PDE for fast likelihood computation and obtained a time reduction of 4% compared to a baseline system. For further efficiency, they have proposed a best mixture prediction (BMP) approach used along with PDE. Based on the strong correlation of successive speech observations, the BMP uses best Gaussian for the last processed observation in order to predict the best Gaussian for the current observation. In evaluating the likelihood of a Gaussian for the current observation, starting the computation on the predicted Gaussian brings a chance to get a high likelihood immediately. Thus, likelihood computation on the next Gaussian mixtures could be largely truncated with PDE. We have performed a statistical analysis of the BMP approach on the HIWIRE corpus, as shown in Table 2. It can be seen that in 52.70% of the cases, the BMP predicts the best Gaussian for the current observation.

<table>
<thead>
<tr>
<th>Rank of Gaussian</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contribution</td>
<td>85.89</td>
<td>10.35</td>
<td>2.40</td>
<td>0.77</td>
<td>0.30</td>
</tr>
<tr>
<td>Rank of Gaussian</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
</tr>
<tr>
<td>Contribution</td>
<td>0.13</td>
<td>0.07</td>
<td>0.04</td>
<td>0.02</td>
<td>0.01</td>
</tr>
</tbody>
</table>

2.2. VQ-based Gaussian selection

If we consider the Table 1 we can see that, in average, 5% of the Gaussians of a GMM contribute with more than 99% of the total probability of the GMM. The likelihood computation could be restricted to the processing of only the closest Gaussians (in each GMM) to the observation under consideration. The main idea of VQ-based Gaussian selection techniques is to predict the best Gaussians of each GMM for a given observation [1, 5, 7]. Usually, VQ-based Gaussian selection is based on an acoustic space clustering. Upon the completion of the training of acoustic models, for each pair of GMM (HMM model state) and acoustic space cluster, a short list of Gaussians is set up. These short lists comprise the \( L \) closest Gaussians to the centroids of the cluster according to a certain distance measure. The cardinality \( L \) of the lists is a parameter depending on the application and the targeted performance.

During recognition, each incoming speech observation is quantized into one acoustic space cluster. The likelihood of each GMM is then computed only on the short list of Gaussians corresponding to the couple of GMM and acoustic cluster. Thus, VQ-based Gaussian selection approaches are essentially two pass techniques based on an offline acoustic space clustering and precomputed best Gaussians lists.

While VQ-based Gaussian selection techniques might perform better in terms of time reduction and likelihood approximation accuracy, they imply a significant growth in storage and memory requirements to store the short lists.

3. Extended PDE

As explained in section 2.1, PDE approximates the likelihood of a GMM for an observation \( x \) with the likelihood of the best Gaussian. The likelihood computation is reduced to a VQ quantisation problem. A proportion of the calculations for sub-optimal Gaussians is truncated whenever the cumulative distortion between such Gaussians and the observation drops below the best distortion obtained so far.

The method we propose in this article is a similar scheme to the PDE approach plus BMP. The main guide line of our method is to eliminate the likelihood computation for a sub-optimal Gaussian faster than the PDE scheme, based on an approximation in the elimination criterion. For the likelihood computation process of a Gaussian \( G \) for an observation \( x \), we modify the
comparison of the cumulative partial distortion $D_k(x|G)$ with the best distortion obtained so far $\tilde{D} = D(x|\tilde{G})$. Rather, we propose to compare $D_k(x|G)$ to an earlier partial distortion between $\tilde{G}$ and $x$: $D_{k+i}(x|\tilde{G})$, where $l$ is a look-ahead rank. For ranks $N - l < k \leq N$, $D_k(x|G)$ is compared to the best distortion $\tilde{D} = D(x|\tilde{G})$. With this criterion approximation, we aim at halting the computations for sub-optimal Gaussians at an earlier stage. This EPDE algorithm combined with BMP is described below.

**Algorithm: Extended PDE**

**Input:**
- $x$: an N-dimensional observation
- $\sum^d_{i=1} \omega_i R(\mu_i, \Sigma_i)$: a GMM with $d$ mixtures
- $B$: index of the best mixture for the last observation
- $l$: a look-ahead parameter

**Output:**
- $\tilde{D}$: the likelihood of the best mixture
- $\tilde{B}$: index of the best mixture for $x$

**Variables:**
- $D_\Phi$: table of partial distortions for the best Gaussian
- $D_\Phi^*$: table of partial distortions for the current Gaussian

**BEGIN**
- $D_\Phi \leftarrow \log(\omega_B) + Z_B$
- For $k = 1$ to $N$
  - $D_\Phi^* \leftarrow D_k - 1 - \frac{1}{l} \frac{(x_k - \mu_k)^2}{\Sigma_{ik}}$
- End For
- $\tilde{D} \leftarrow D_N$ ; $\tilde{B} \leftarrow B$
- For $i = 1$ to $d$
  - Do
    - If ($i = B$) Then Skip to next Gaussian
    - $D_\Phi \leftarrow \log(\omega_i) + Z_i$
    - For $k = 1$ to $(N - l)$
      - $D_\Phi^* \leftarrow D_k - 1 - \frac{1}{l} \frac{(x_k - \mu_k)^2}{\Sigma_{ik}}$
    - End For
    - If ($D_\Phi < D_\Phi^*$) Then Skip to next Gaussian
  - End Do
- For $k = (N - l + 1)$ to $N$
  - $D_\Phi \leftarrow (N - l + 1) - \frac{1}{l} \frac{(x_k - \mu_k)^2}{\Sigma_{ik}}$
- If ($D_\Phi < D_N$) Then Skip to next Gaussian
- End For
- $D \leftarrow D_N$ ; $\tilde{B} \leftarrow i$
- Switch tables: $D_\Phi \leftrightarrow D_\Phi^*$
- End For
- Return($\tilde{D}, \tilde{B}$)

**END**

While PDE selects the closest Gaussian to the observation, EPDE does not guarantee this optimal selection. Indeed, one can see that there could exist situations where EPDE can discard the actual best Gaussian. The “real” best Gaussian $\tilde{G}$ (in a GMM) for an observation $x$ could satisfy $D_k(x|\tilde{G}) < D_{k+i}(x|\tilde{G})$ (at a partial likelihood computation step $k$). In EPDE, the constraint of optimality is relaxed in favor of better computation time reduction. We can see in Figure 1 the error in Gaussian selection introduced by EPDE depending on the look-ahead parameter $l$. In this figure, we notice that narrower look-ahead values imply poorer decision on the best Gaussian. For a value of $l = 3$, EPDE picks the closest Gaussian to the observation in only $43.2\%$ of the cases. A value of look-ahead of $l = 13$ (rep. $16, 20$ and $25$) results in better selection accuracy of $96.7\%$ (rep. $98.7\%$, $99.7\%$ and $99.97\%$).

**Figure 1:** Histogram of the rank of the selected Gaussian by EPDE, with different look-ahead parameters. (conditions 1).

### 4. Combined EPDE and DGS

In [4] we have presented a new method of dynamic Gaussian selection (DGS) for fast likelihood computation. This method is based on the framework of PDE and aims at minimizing the recognition accuracy introduced by the likelihood approximation in PDE. For each GMM, DGS dynamically selects a list of closest Gaussians to the observation vector. Contrary to PDE, the likelihood of a GMM is approximated with the sum of the likelihoods of Gaussians in the dynamically selected list.

DGS uses the rank $\hat{k}$ (see 2 and 2.1) at which the PDE stops the likelihood computation of a Gaussian $G$, as a hint about the distance between the observation $x$ and $G$. If the stop rank $\hat{k}$ is less than a certain threshold $\gamma$, the Gaussian $G$ is considered very far from $x$ compared to other Gaussians in the GMM; namely the best Gaussian found so far. On the other hand, if $\hat{k}$ is greater than $\gamma$, then $G$ is considered close to $x$ and is selected to contribute in the total likelihood of the GMM. In this case, the likelihood computation of $G$ is resumed (from the rank $\hat{k}+1$) and the full likelihood $l(x|G)$ is calculated and added to the total likelihood of the GMM. The value of $\gamma$ is closely related to the dimensionality $N$ of the models, and it should be $70\%$ to $90\%$ of $N$. It should be chosen according to the targeted accuracy and speed of the recognition.

Our experiments in [4] show that DGS introduces less recognition accuracy degradation than PDE at the expense of a slight computational time increase. For that matter, we have combined DGS and EPDE for faster likelihood computation with minimal accuracy decrease. The procedure of EDGS (DGS + EPDE) is basically the same as described in the last paragraph, except that EPDE is the framework.

### 5. Experiments and discussion

We have tested the EPDE and EDGS methods on the test part of TIMIT corpus. As we are interested mainly in the time performance of the methods, the tests we have carried out are phonetic level recognitions. The acoustic models we have used are 3-state HMM mono-phones trained on TIMIT. We chose a parametrisation of $13$ MFCC coefficients ($12 + \text{energy}$) with their first and second derivatives.

We investigated the values of the look-ahead $l \in \{3, 5, 7, 10, 13, 16, 20\}$ for EPDE (see section 3). For EDGS, we chose a value of the threshold $\gamma = 35$ (see section 4). Table 3 summarises the results of the tests with different values of $l$. For comparison purposes, this table also
comprises the results of PDE and DGS approaches.

As can be seen in Table 3, PDE and DGS achieve around 17%-23% likelihood computational time reduction. We can see that DGS has virtually the same accuracy as the baseline system. PDE is about 5% faster (relative) than DGS, at the expense of about 0.5% recognition accuracy degradation (relative). Both PDE and DGS reduce the computations by 70.5% for 128 mixtures models (and 66.5% for 64 mixtures).

As expected, a narrower look-ahead value \( l \) results in a faster likelihood calculation and a greater reduction in the computations count. Besides, smaller values of \( l \) deteriorate the recognition accuracy. Comparing EPDE to PDE, with 128 (resp. 64) mixtures per state, a value of \( l = 20 \) decreases the recognition time of about 2.2% (resp. 1.3%) and reduces the number of computations by 13% (resp. 12.5%), with virtually the same recognition accuracy for both PDE and EPDE. We observe the same phenomena when comparing EDGS to DGS: with 128 (resp. 64) mixtures, a value of \( l = 20 \) decreases the recognition time of about 3.8% (resp. 3%) and reduces the number of computations by 13.6% (resp. 12.5%), with virtually the same recognition accuracy. Furthermore, we can see that for a value of \( l = 13 \), EDGS tested on 128 mixtures models (resp. 64) performs 7.3% (resp. 5.7%) faster than PDE, and requires 29.8% (resp. 28.5%) less computations, with virtually the same recognition accuracy.

### EPDE to DGS

By using an extended PDE (EPDE) approach, we can achieve a speedup of about 5% (relative) compared to PDE. This EPDE approach relaxes the constraint of choosing the closest Gaussian in the GMM to the observation vector in favor of faster processing. This speedup is achieved through comparing the partial distortion (between a Gaussian and an observation) to a partial distortion of the best Gaussian, rather than the final distortion of the latter. Results show that EPDE performs faster and requires less computations than PDE, for the same error rate. We have also combined EPDE with a dynamic Gaussian selection technique (DGS). At the same error rate, the combination of DGS and EPDE is 7.3% faster than PDE and requires 29.8% less computations. This makes it suitable for time consuming speech recognition applications. Especially, applications with limited resources or where memory access is highly penalizing -such as mobile platforms- could benefit from these approaches.

### 6. Conclusions

In this article, we have presented a new approach for fast likelihood computation which is based on a similar scheme as PDE. This extended PDE (EPDE) approach relaxes the constraint of choosing the closest Gaussian in the GMM to the observation vector in favor of faster processing. This speedup is achieved through comparing the partial distortion (between a Gaussian and an observation) to a partial distortion of the best Gaussian, rather than the final distortion of the latter. Results show that EPDE performs faster and requires less computations than PDE, for the same error rate. We have also combined EPDE with a dynamic Gaussian selection technique (DGS). At the same error rate, the combination of DGS and EPDE is 7.3% faster than PDE and requires 29.8% less computations. This makes it suitable for time consuming speech recognition applications. Especially, applications with limited resources or where memory access is highly penalizing -such as mobile platforms- could benefit from these approaches.

### 7. References


