NOISE COMPENSATION BY A SEQUENTIAL KULLBACK PROXIMAL ALGORITHM

Kaisheng Yao[⋆]*, Kuldip K. Paliwal*[⋆†]*, Bertram E. Shi*[†]*, and Satoshi Nakamura*[⋆]

[⋆]ATR Spoken Language Translation Research Laboratories
2-2, Hikaridai Seika-cho, Souraku-gun, Kyoto, 619-0288, Japan
[†] School of Microelectronics Engineering, Grif£th University, Australia
[†]Department of Electrical and Electronic Engineering
Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong

ABSTRACT

We present sequential parameter estimation in the framework of the Hidden Markov Models. The sequential algorithm is a sequential Kullback proximal algorithm, which chooses the Kullback-Liebler divergence as a penalty function for the maximum likelihood estimation. The scheme is implemented as £lters. In contrast to algorithms based on the sequential EM algorithm, the algorithm has faster convergence rate in parameter estimation, and the computational complexity is proportional to the algorithms based on the sequential EM algorithm. In particular, we derive sequential noise parameter estimation for a model-based sequential noise compensation method for nonstationary noise environments. Noise parameter estimation, updating and speech recognition are carried out frame by frame. Simulation results have shown that the derived schemes can have faster convergence rate than the sequential noise compensation based on the sequential EM algorithm.

1. INTRODUCTION

Speech recognition is considered as the problem of choosing a corresponding state sequence $\mathbf{q} = (q_1 q_2 \cdots q_T)$ with the maximum joint likelihood $P(\mathbf{q}, Y_T)$ of the observation sequence Y_T given model parameter Θ. For speech recognition in matched condition, Θ can be obtained during the training stage. However, for speech recognition in mismatched condition, Θ or the features for speech recognition have to be transformed to decrease the mismatch. Noise is one of the major sources for mismatch. Noise compensation has been considered to be essential for speech recognition in real environments. Among the approaches for noise compensation, model-based approach assumes an explicit model representing noise effects on speech features or models. For example, a non-linear transformation of mean vector in clean speech models can be carried out on the mean μ^l_{imj} of each mixture m at state i in clean speech models after estimation of a noise parameters μ^l_{nj} in the log-spectral £lter bank j [1],

$$\hat{\mu}^l_{imj} = \mu^l_{imj} + \log(1 + \exp(\mu^l_{nj} - \mu^l_{imj})) \qquad (1)$$

where $1 \leq j \leq J$, and J is the total number of log-spectral £lter banks. Superscript l indicates parameters are in the log-spectral domain. This function assumes that the noise variance is very small, and accordingly, only the mean of the acoustic models are transformed. Other model based methods such as[2] employ similar functional formula with different complexities. Most of the methods assume that the noise statistics are constant, so that noise parameter can be estimated in advance, and plugged into the noise compensation procedures.

In this paper, we consider a sequential noise compensation scheme for nonstationary noise case. Here, the noise statistics are varying during the speech recognition. Thus, the model parameter is $\Theta(t)$. Our motivations of this work are from these considerations. Firstly, correct noise estimation is hardly obtainable in situations such as push-to-talk working scenarios. If the noise parameters were wrong, the recognition performance would be very poor. Secondly, even though methods such as Maximum Likelihood Linear Regression (MLLR) [3] can be adopted to transform the acoustic models before recognition in mismatched condition, it requires adaptation data. Once the transformation has been £nished, the noise during recognition is still assumed to be stationary. Thus, the methods can not adapt models to unseen noise during recognition. Thirdly, previous algorithms [3] work utterance by utterance. The convergence rate is found to be slow. We believe algorithms working frame by frame may have faster convergence rate than those based on adaptation in an utterance level.

Our sequential noise compensation works frame by frame. In each frame, it £rstly propagate the joint likelihood $P(\mathbf{q}, \Theta(t)$ till frame t, and then update estimation of $\Theta(t+1)$ via the principle of maximum likelihood estimation, where the likelihood is given as $\sum_{\mathbf{q}} P(\mathbf{q}; \Theta(t+1))$. The joint likelihood propagation and parameter updating are carried out in an iterative way.

Due to dif£culties in directly calculating the likelihood in the HMM framework, methods based on the sequential Expectation Maximization [4] can be adopted for the purpose. However, as is well known, the EM algorithm suffers from slow convergence rate. For sequential adaptation, faster convergence rate is always desirable. Recently, Chréten and Hero III [5] have introduced a Kullback proximal algorithm. The algorithm is a class of accelerated EM algorithm for likelihood function maximization via exploiting of a general relation between EM and proximal-point algorithms. These algorithms converge and can have quadratic rates of convergence even with approximate updating.

In this paper, we derive sequential algorithm based on the Kullback proximal point algorithm for noise parameter estimation. The noise compensation is carried out via a Log-Add noise compensation method [1], which is shown in Equation (1). The contributions of this paper are as follows: 1), In contrast to the sequential algorithm based on the EM algorithm, the derived sequential algorithm is shown to have faster convergence rate than the algorithm based on the sequential EM algorithm. This is achieved by proper choice of a relaxation factor, β_t. 2), The sequential EM algorithm

is shown to be a particular case of the sequential Kullback proximal point algorithm, the case of setting $\beta_t = 1.0$. 3), In particular, we show the application of the sequential Kullback proximal point algorithm in the sequential noise compensation. Experiments carried out so far have con£rmed the advantage of the proposed algorithm over the algorithm based on the sequential EM algorithm, in terms of convergence rate and recognition performance in mismatched signal-to-noise ratio (SNR) conditions.

2. BACKGROUND

During speech recognition, joint likelihood of observation sequence Y_t and state sequence $\mathbf{q}$ are propagated via Viterbi approximation, which is given as,

$$\alpha_t(i;\theta) = \alpha_{t-1}(\zeta^\star;\theta)a_{\zeta^\star i}b_i(y_t) \tag{2}$$

where $q_t = i, \zeta^\star = \arg\max_\zeta \alpha_{t-1}(\zeta;\theta)a_{\zeta i}$. $b_i(y_t)$ is the emission probability of y_t at state i given parameter θ. Sequential parameter estimation during the speech recognition is carried out via maximizing the log-likelihood, i.e.,

$$\theta(t) = \arg\max_{\theta\in R^J} l_t(\theta)$$

where $l_t(\theta)$ is the log-likelihood till frame t, given as,

$$l_t(\theta) = \log\sum_{i=1}^{N}\alpha_t(i;\theta)$$

where N is the total number of states. Direct calculation on the likelihood is dif£cult in the HMM frameworks, thus, the sequential Expectation Maximization (EM) algorithm is a popular algorithm [4] to £nd the ML estimation of $\theta(t)$.

The algorithm has the conditional expectation of the log likelihood of the complete data $\mathbf{q}, Y_t$ until time t, i.e., $Q_t(\Theta(t-1),\theta) = E[\log f(\mathbf{q}, Y_t)|Y_t, \Theta(t-1)]$. $\Theta(t-1) = (\theta(1)\theta(2)\cdots\theta(t-1))$ denotes the sequence of estimated parameters in HMMs till time $t-1$ based on the observation sequence Y_{t-1}. By maximizing a second order Taylor series approximation of the $Q_t(\Theta(t-1),\theta)$ by $Q_t(\Theta(t-1),\theta(t-1))$, we can obtain the following updating formula,

$$\theta(t) = \theta(t-1) + (R_t(\theta(t-1)))^{-1}S(\theta(t-1), y_t) \tag{3}$$

where $S(\theta(t-1), y_t) = \frac{\partial Q_t(\Theta(t-1),\theta)}{\partial\theta}|_{\theta=\theta(t-1)}$, and $R_t(\theta(t-1))$ is with element of $-\frac{\partial^2 Q_t(\Theta(t-1),\theta)}{\partial\theta^2}|_{\theta=\theta(t-1)}$.

Alternatively, maximum likelihood estimation can also be addressed by the Kullback proximal point algorithm [5].

Proposition 1 *The sequential EM algorithm is equivalent to the following recursion with* $\beta_t = 1, t = 1, 2, ...,$

$$\theta(t) = \arg\max_{\theta\in R^J}\{l_t(\theta) - \beta_t I_y(\theta, \Theta(t-1))\} \tag{4}$$

where $I_y(\theta,\Theta(t-1)) = \sum_q \log\frac{f(\boldsymbol{q}|Y_t;\Theta(t-1))}{f(\boldsymbol{q}|Y_t;\theta)} f(\boldsymbol{q}|Y_t;\Theta(t-1))$ *is the Kullback-Liebler (K-L) divergence.*

Note that our proposition is a sequential version of the batch version in Proposition 1 in [5]. The proposition can be similarly proved as in [5].

The Kullback-Liebler (KL) divergence between successive iterates of the posterior densities of the complete data works as a regulization factor. Following the terminology in [5], we denote the above updating procedure as the sequential Kullback proximal algorithm.

3. THE ALGORITHM

The proposition 1 gives the following sequential algorithm:

Let $\theta(0)$ be the initial model estimate. Then given y_t at time t, the recursive update of $\theta(t)$ is given as,

$$\begin{aligned}\theta(t) &= \theta(t-1) \\ &- \frac{\frac{\partial Q_t(\Theta(t-1);\theta)}{\partial\theta}|_{\theta=\theta(t-1)}}{\beta_t\frac{\partial^2 Q_t(\Theta(t-1),\theta)}{\partial\theta^2}|_{\theta=\theta(t-1)} + (1-\beta_t)\frac{\partial^2 l_t(\theta)}{\partial\theta^2}|_{\theta=\theta(t-1)}}\end{aligned} \tag{5}$$

where

$$\frac{Q_t(\Theta(t-1),\theta)}{\partial\theta}|_{\theta=\theta(t-1)} = \sum_{i=1}^{N}\gamma_t(i;\theta(t-1))\frac{\partial\log b_i(y_t)}{\partial\theta}|_{\theta=\theta(t-1)} \tag{6}$$

$$\begin{aligned}\frac{\partial^2 Q_t(\Theta(t-1),\theta)}{\partial\theta^2}|_{\theta=\theta(t-1)} &= \frac{\partial^2 Q_{t-1}(\Theta(t-2),\theta)}{\partial\theta^2}|_{\theta=\theta(t-1)} \\ &+ \sum_{i=1}^{N}\gamma_i(t;\theta(t-1))\frac{\partial^2\log b_i(y_t)}{\partial\theta^2}|_{\theta=\theta(t-1)}\end{aligned} \tag{7}$$

where $\gamma_i(t;\theta)$ is the posterior probability at state i given observation Y_t and model parameter θ. For sequential updating, we only consider the £ltering scheme, and accordingly, the posterior probability is given as,

$$\gamma_t(i;\theta) = \frac{\alpha_t(i;\theta)}{\sum_{\eta=1}^{N}\alpha_t(\eta;\theta)} \tag{8}$$

3.1. Implementation

Exact calculation of the second order differentiation of $l_t(\theta)$ is computationally expensive. We propose an approximation, in which the value is approximated at θ around $\theta(t-1)$ as,

$$\begin{aligned}\frac{\partial^2 l_t(\theta)}{\partial\theta^2} &= \sum_{i=1}^{N}\frac{\partial\gamma_i(t;\theta)}{\partial\theta}\frac{\partial\log b_i(y_t)}{\partial\theta} \\ &+ \sum_{i=1}^{N}\gamma_i(t;\theta)\frac{\partial^2\log b_i(y_t)}{\partial\theta^2}\end{aligned} \tag{9}$$

After some mathematical derivation, we have,

$$\begin{aligned}\frac{\partial^2 l_t(\theta)}{\partial\theta^2} &= \sum_{i=1}^{N}\sum_{m=1}^{M}\gamma_t(i,m;\theta)[(\frac{\partial\log b_{im}(y_t)}{\partial\theta})^2 + \frac{\partial^2\log b_{im}(y_t)}{\partial\theta^2}] \\ &- (\sum_{i=1}^{N}\sum_{m=1}^{M}\gamma_t(i,m;\theta)\frac{\partial\log b_{im}(t)}{\partial\theta})^2\end{aligned} \tag{10}$$

where $b_{im}(y_t)$ is the emission probability of y_t at state i and mixture m given model parameter $\theta(t)$. c_{im} is the Gaussian mixture weight, and $\sum_{m=1}^{M} c_{im} = 1$. $b_i(y_t) = \sum_{m=1}^{M} c_{im}b_{im}(y_t)$ and $\gamma_t(i,m;\theta) = \gamma_t(i;\theta)\frac{c_{im}b_{im}(y_t)}{b_i(y_t)}$.

3.2. Remarks

Equation (5) shows that the β_t works as a balance between an accumulated estimation by $\frac{\partial^2 Q_t(\Theta(t-1);\theta)}{\partial\theta^2}$ and the current estimation $\frac{\partial^2 l_t(\theta)}{\partial\theta^2}$ at time t. Decreasing β_t from 1.0 to zero will make the updating biased to the current estimation. When $\beta_t = 0.0$, the updating is a Newton step, which has superlinear convergence rate but may be divergent when the procedure was wrongly initialized. Increasing β_t to in£nity will £x the parameter estimation without adaptation to new data input. The smaller the β_t, the faster the convergence rate of the algorithm.

We see that the parameter updating by the sequential EM algorithm corresponds to setting $\beta_t = 1.0$ in the Equation (5).

Similar to [4], the exponential forgetting can be adopted, where the forgetting factor $0 < \rho \leq 1.0$.

3.3. Application to the noise parameter estimation

Speci£cally, for $\theta(t) = \mu^l_{nj}(t)$, the noise parameter updating requires the following calculations in each mixture m at state i. i.e.,

$$\frac{\partial \log b_{im}(y_t;\theta)}{\partial\theta} = \sum_{k=1}^{K}[d_{kj}\frac{(y_t(k)-\mu_{imk}(t-1))}{\sigma^2_{imk}}\frac{\partial\mu^l_{imj}(t)}{\partial\theta}]$$

$$\frac{\partial^2 \log b_{im}(y_t;\theta)}{\partial\theta^2} = \sum_{k=1}^{K}[-\frac{1}{\sigma^2_{imk}}d^2_{kj}(\frac{\partial\mu^l_{imj}(t)}{\partial\theta})^2 + \frac{y_t(k)-\mu_{imk}}{\sigma^2_{imk}}d_{kj}\frac{\partial^2\mu^l_{imj}(t)}{\partial\theta^2}]$$

where $\frac{\partial\mu^l_{imj}(t)}{\partial\theta}$ and $\frac{\partial^2\mu^l_{mj}(t)}{\partial\theta^2}$ are given as $\frac{\exp(\mu^l_{nj}(t)-\mu^l_{imj})}{1+\exp(\mu^l_{nj}(t)-\mu^l_{imj})}$ and $\frac{1}{(1+\exp(\mu^l_{nj}(t)-\mu^l_{imj}))^2}$, respectively. $\mu_{imk}(t-1)$ is the corresponding compensated cepstral mean at cepstral k, obtained after the Discrete Cosine Transform (DCT) of $\{\mu^l_{mj}(t-1) : 1 \leq j \leq J\}$. σ^2_{imk} is the diagonal variance at cepstral index k in mixture m at state i. $y_t(k)$ is the cepstral observation element at cepstral index k in the observation vector. d_{kj} is the DCT coef£cient.

In fact, the algorithm works in a vector form for noise parameter updating. For simplicity in expression, we adopt the scalar formula. Also, the parameter updating is carried out once at each frame, instead of several iterations at each frames until convergence. We speculate that the latter approach may make the convergence rate of parameter estimation even faster than what we have obtained in the experiments.

4. EXPERIMENTS

4.1. Experimental setup

Experiments were performed on the TI-Digits connected digits database, which was down-sampled to 16kHz. Five hundred clean speech utterances from 15 speakers and 100 utterances from four speakers unseen in the training set were used for training and testing, respectively. Digits and silence were respectively modeled by 10-state and 3-state (including a non-emitting initial and £nal state) whole word HMMs with 4 diagonal Gaussian mixtures in each state. Contaminated speech was generated by arti£cially adding different levels of noise to the clean speech. Noises were White noise. The signal-to-noise ratio (SNR) was measured by SNR = $10\log_{10}$(energy of the total clean speech utterances/energy of additive noise). Noise statistics were modeled by a single Gaussian mixture.

The window size was 25.0ms with a 10.0ms shift. Twenty-six £lters were used in the binning stage, i.e., $J = 26$. The features were MFCC + Δ MFCC. The baseline system had a 1.3% Word Error Rate (WER) under clean conditions.

The sequential algorithms for the sequential noise compensation is un-supervised, i.e., we assume no knowledge of the actual transcription. Before recognition of a set of utterances, $\frac{\partial^2 Q_t(\Theta(t-1),\theta)}{\partial\theta^2}$ was initialized to be the minus 100 times of the variance of noise at each log-spectral £lter bank. $\frac{\partial Q_t(\Theta(t-1),\theta)}{\partial\theta}$ was initialized to zero. The noise statistics were estimated from 5 seconds of noise before the recognition process. During recognition of this set of utterances, the parameters obtained from the last frame of each utterance in the set are used to initialize the updating procedure in Equation (5) for the next utterance. The forgetting factor ρ was set to 0.995.

4.2. Experimental results

Experiment results are shown in Table 1. Baseline system without noise compensation and the system with the Log-Add noise compensation in known noise condition are shown in the second and third row. As can be seen from the tables, the Log-Add noise compensation is effective in improving system robustness to noise.

We then test the system performance in unknown noise condition. As said before, the present work is motivated by adapting system to unknown working environments, or adapting the system to the environments where the noise parameter is hardly obtainable. Our method to evaluate system performance is by making the SNRs in the environment for the noise parameter estimation (initialization) and the SNRs in the environments for testing the systems different. The SNRs for noise parameter initialization is shown in the right most column. The SNRs for testing are in the £rst row.

WSA denotes system with the Log-Add noise compensation but without sequential adaptation schemes. SEM and SKP denote system with sequential noise compensation by the sequential EM algorithm and the sequential Kullback proximal algorithm, respectively. In our experiments, we set $\beta_t = 0.9$ for the sequential Kullback proximal algorithm.

Observing the tables, we notice that when initialization and testing are in the same SNR conditions, the performance of the sequential noise compensations is around that of the system compensated by the Log-Add method with known noise statistics, labeled LAdd.

For situations where system operated in mismatched SNR conditions, experimental results show that the system with sequential noise compensation can effectively compensate contaminating noise in unknown SNR conditions, whereas the system without it performs poorly. For example, in 16.0dB White noise, "SEM" and "SKP" have 19.3% and 19.0% WER when initialized at 20.4dB White noise, whereas the same (initial) noise parameter make the system "WSA" without sequential compensation have performance at 55.3% WER.

Observations from the tables also show that, in most cases, SKP with sequential Kullback proximal algorithm has lower WER than the SEM with sequential EM algorithm. This con£rms our

Table 1. WER (in %) of the system in White noise. Baseline is the system without noise compensation. LAdd denotes system with known noise parameter compensated. SEM and SKP each denote the system with sequential noise compensation by the sequential EM algorithm and the system with sequential noise compensation by the sequential Kullback proximal algorithm. WSA denotes system with the Log-Add noise compensation with noise parameters as the initial parameters for the systems SEM and SKP but without sequential adaptation. The central four columns denote the testing SNR. The right most column denotes the SNR in which the initial noise parameters were estimated.

SNR (dB)	8.8	16.0	20.4	40.4	Initial
Baseline	81.3	62.0	37.3	22.3	
LAdd	32.0	14.7	2.0	2.7	Matched
WSA	32.0	21.3	46.0	51.7	8.8
SEM	34.0	16.7	9.0	10.7	8.8
SKP (β_t = 0.9)	34.0	16.7	7.7	10.0	8.8
WSA	38.7	14.7	38.3	32.0	16.0
SEM	34.0	15.7	4.0	6.3	16.0
SKP (β_t = 0.9)	34.0	15.7	4.0	5.7	16.0
WSA	61.3	55.3	2.0	2.3	20.4
SEM	45.7	19.3	2.3	2.3	20.4
SKP (β_t = 0.9)	40.3	19.0	2.7	2.0	20.4
WSA	66.7	52.3	2.3	2.7	40.4
SEM	45.3	19.7	2.7	2.7	40.4
SKP (β_t = 0.9)	46.0	20.0	2.0	2.0	40.4

remark in section 3, where we point out that a comparatively faster convergence rate can be achieved by setting β_t smaller than 1.0.

To explicitly show the convergence rate in parameter estimation, we de£ne the Mean Square Error (MSE) of the noise compensation procedure at each frame t as,

$$MSE(t) = \frac{1}{J}\sum_{j=1}^{J}(\mu_{nj}^{l}(t) - \hat{\mu}_{nj}^{l})^2$$

where $\mu_{nj}^{l}(t)$ and $\hat{\mu}_{nj}^{l}$ denote the sequentially estimated noise parameter and the known mean of the contaminating noise parameter, respectively.

Figure 1 shows the MSEs of the sequential Kullback proximal algorithm. The sequential Kullback proximal algorithm are with $\beta_t = 10.0, 2.0, 1.0, 0.9$ and 0.5. They were initialized at 8.8dB and tested in 16.0dB SNR condition. The testing utterances are the testing set contaminated by White noise. It shows that the convergence rate of the sequential Kullback proximal algorithm can be controlled by β_t. The larger the β_t, the smaller the convergence rate. It con£rms that the convergence rate can be faster when $\beta_t < 1.0$ compared with the sequential EM algorithm. The £gure also shows that, after convergence, the sequential Kullback proximal algorithm with smaller β_t may have larger estimation error than that with larger β_t. They show similar trends when initialized and tested in other SNR conditions.

We have also conducted experiments in Babble noise. All the experiments carried out so far have similar trends in performance and convergence rate.

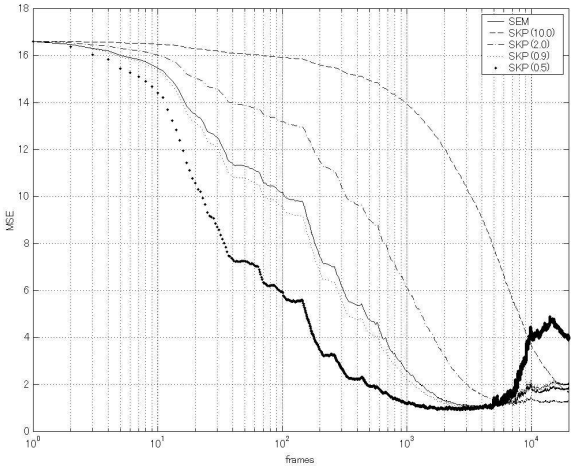

Fig. 1. MSE of the sequential noise compensation in unknown SNR condition ($\rho = 0.995$). Horizontal axis is the frame index in logarithm scale. Sequential Kullback proximal algorithm has β_t set to be 10.0, 2.0, 1.0, 0.9, and 0.5. Systems are initialized at 8.8 dB White noise and tested at 16.0 dB White noise.

5. CONCLUSIONS

We have presented a sequential parameter estimation method for maximum likelihood estimation based on the Kullback proximal algorithm. The method shows faster convergence rate than that based on the EM algorithm by controlling a relaxation factor β_t. We have applied the method to sequential noise compensation where noise is nonstationary and noise estimation is carried out during speech recognition.

6. REFERENCES

[1] K. Yao, B. E. Shi, S. Nakamura, and Z. Cao, "Residual noise compensation by a sequential em algorithm for robust speech recognition in nonstationary noise," in *ICSLP*, 2000, vol. 1, pp. 770–773.

[2] M.J.F.Gales and S.J.Young, "Robust speech recognition in additive and convolutional noise using parallel model combination," *Computer Speech and Language*, vol. 9, pp. 289–307, 1995.

[3] P. C. Woodland, M. J. F. Gales, and D. Pye, "Improving environmental robustness in large vocabulary speech recognition," in *ICASSP*, 1996, pp. 65–68.

[4] V. Krishnamurthy and J. B. Moore, "On-line estimation of hidden markov model parameters based on the kullback-leibler information measure," *IEEE Trans. on Signal Processing*, vol. 41, no. 8, August 1993.

[5] S. Chréten and A. O. Hero III, "Kullback proximal point algorithms for maximum–likelihood estimation," *IEEE. Trans. on IT*, vol. 46, no. 5, August 2000.