Maximum likelihood sub-band adaptation for robust speech recognition

Donglai Zhu, Satoshi Nakamura, Kuldip K. Paliwal, Renhua Wang

Reporter:邱聖權

Professor:陳嘉平

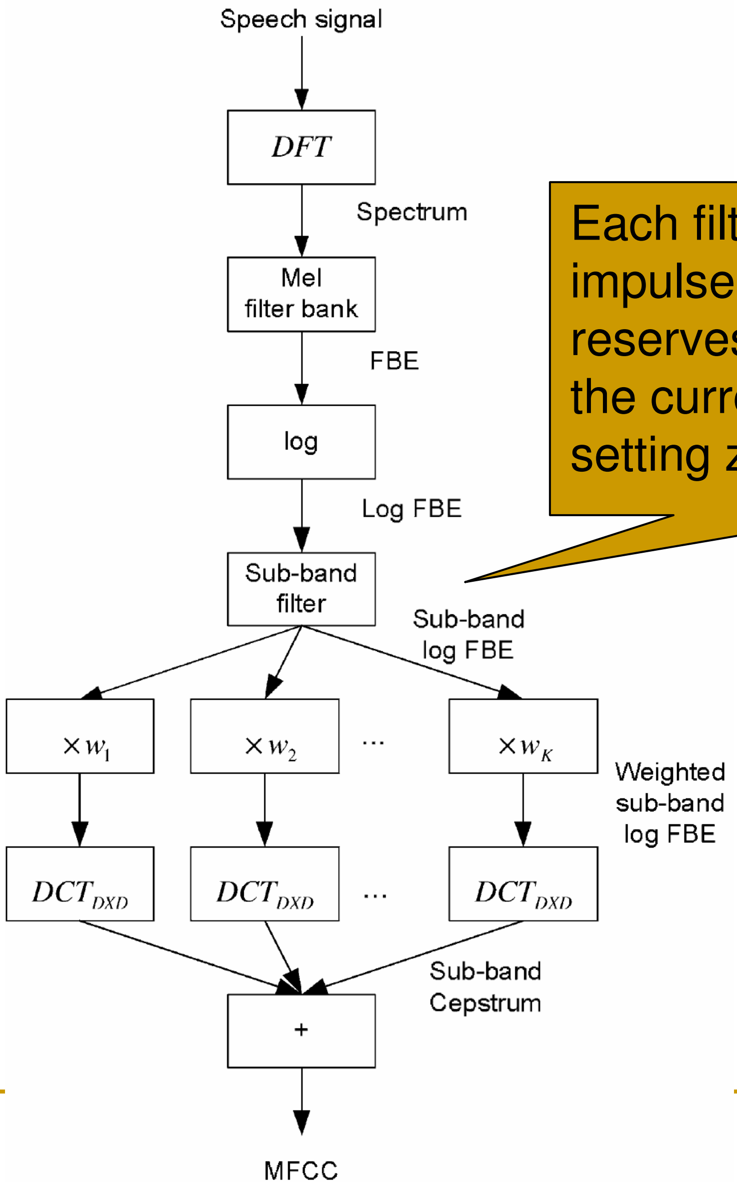
Introduction

- In human auditory perception, the linguistic message gets decoded independently in different frequency sub-bands.
- The final decoding decision is based on merging the decisions from the sub-bands.

Sub-band approach in ASR

- The full-band power spectrum is divided into several sub-bands.
- Features are extracted from the sub-bands spectra independently.
- Features from each sub-bands are combined depending on their reliability by multiplying a weight.
- Sub-band weighting may be performed on the feature space or the modeled space.

Weighting procedure on feature space



Each filter has a unit rectangular impulse response, which reserves the log FBE bins within the current sub-band while setting zero for those outside

Weighting procedure

- The log FBE vector $f = \{f_1, f_2, \ldots, f_D\}$ is separated into K sub-bands $\{\boldsymbol{f}_1, \boldsymbol{f}_2, \ldots, \boldsymbol{f}_K\}$ by the sub-band filter.
- The log FBE for each sub-band is a D-dimensional vector, the sum of these vectors is the original vector: $$\boldsymbol{f} = \sum_{i=1}^{K} \boldsymbol{f}_i$$
- Each sub-bend vector is multiplied by a weight:

$$\hat{\boldsymbol{f}}_i = w_i \boldsymbol{f}_i,\ 1 \le i \le K$$

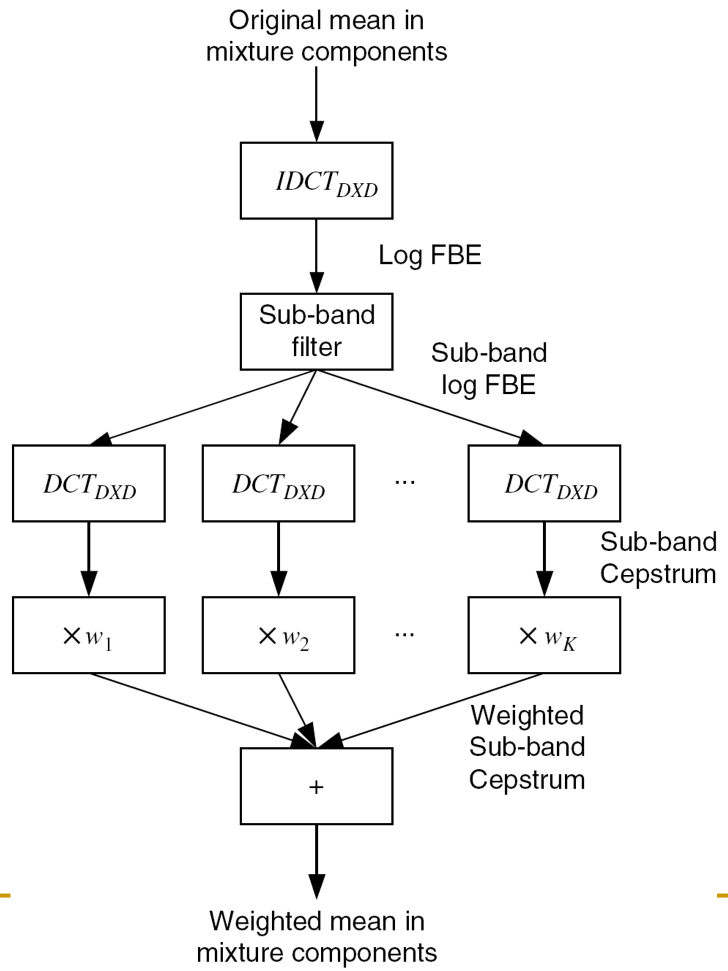
Weighting procedure

- Each sub-band vector is converted to a cepstrum vector via DCT: $\hat{\boldsymbol{c}}_i = \mathrm{DCT}(\hat{\boldsymbol{f}}_i),\ 1 \le i \le \mathrm{K}$
- The final MFCC is obtained by summing all sub-band cepstrum vectors: $\hat{\boldsymbol{c}} = \sum_{i=1}^{K} \hat{\boldsymbol{c}}_i$
- Because DCT is a linear transformation, weighting on the sub-band log FBE vector is identical with that on the sub-band cepstrum vector

$$\hat{\boldsymbol{c}} = \sum_{i=1}^{K} \hat{\boldsymbol{c}}_i = \sum_{i=1}^{K} \mathrm{DCT}(w_i \boldsymbol{f}_i) = \sum_{i=1}^{K} w_i \mathrm{DCT}(\boldsymbol{f}_i) = \sum_{i=1}^{K} w_i \boldsymbol{c}_i$$

Weighting procedure on model space

weights are performed on the mean vectors of Gaussian mixture components in HMMs



Experiment

- Add noise to particular band.
- Scale of bands is known.
- Weight in the noisy band is 0, and the others are 1.

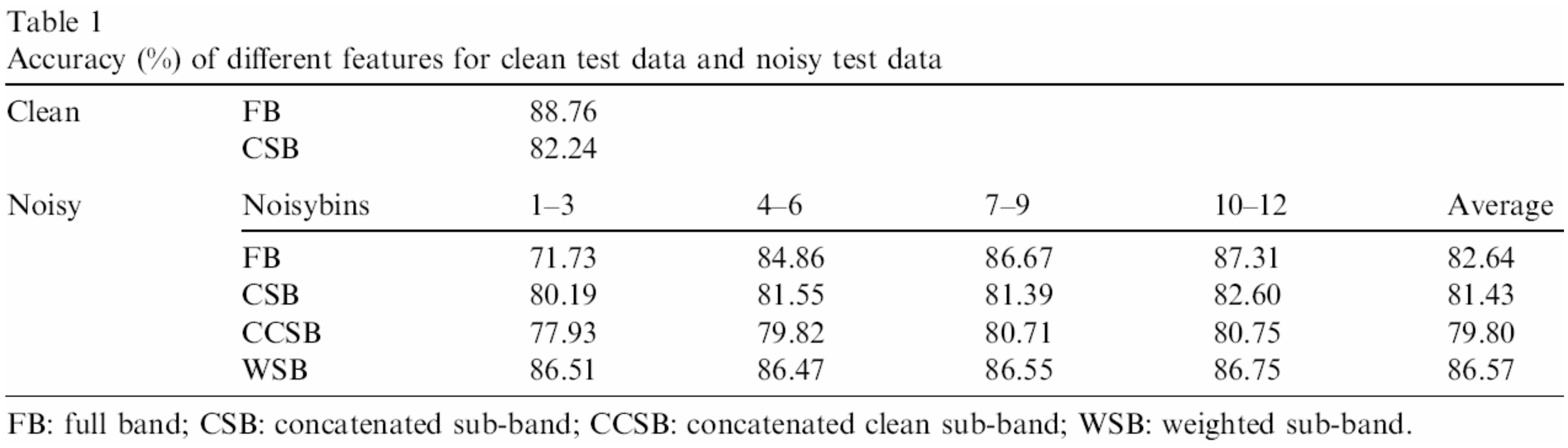


Table 1
Accuracy (%) of different features for clean test data and noisy test data

Clean	FB	88.76				
	CSB	82.24				
Noisy	Noisybins	1–3	4–6	7–9	10–12	Average
	FB	71.73	84.86	86.67	87.31	82.64
	CSB	80.19	81.55	81.39	82.60	81.43
	CCSB	77.93	79.82	80.71	80.75	79.80
	WSB	86.51	86.47	86.55	86.75	86.57

FB: full band; CSB: concatenated sub-band; CCSB: concatenated clean sub-band; WSB: weighted sub-band.

Sub-band weighting adaptation

- Given a trained model ϕ_X, and test data $\boldsymbol{Y}$, a recognizer recognize a word sequence by MAP decoder

$$\hat{\boldsymbol{W}} = \arg\max_{w} P(\boldsymbol{W} \mid \boldsymbol{Y}, \phi_X) = \arg\max_{w} P(\boldsymbol{Y} \mid \boldsymbol{W}, \phi_X) P(\boldsymbol{W} \mid \phi_X)$$

- we assume the distortion between training data $\boldsymbol{X}$ and test data $\boldsymbol{Y}$ is invertible, the inverse function is F_υ, where υ is the function parameters.

$$\boldsymbol{X} = F_\upsilon(\boldsymbol{Y})$$

Feature space adaptation

- Minimizing the mismatch can be done by maximizing the joint likelihood.

$$(\hat{\upsilon}, \hat{\boldsymbol{W}}) = \arg\max_{\upsilon, \boldsymbol{W}} P(\boldsymbol{Y} \mid \boldsymbol{W}, \upsilon, \phi_X) P(\boldsymbol{W} \mid \phi_X)$$

- Because $\boldsymbol{W}$ is a fixed value, we consider υ only.

$$\hat{\upsilon} = \arg\max_{\upsilon} P(\boldsymbol{Y} \mid \upsilon, \phi_X)$$

$$= \arg\max_{\upsilon} \sum_{\boldsymbol{S}} \sum_{\boldsymbol{K}} p(\boldsymbol{Y}, \boldsymbol{S}, \boldsymbol{K} \mid \upsilon, \phi_X)$$

$\boldsymbol{S}$ is the set of all possible state sequences,

$\boldsymbol{K}$ is the set of all mixture component sequences,

Model space adaptation

- Model space is similar to feature space.

$$\hat{\eta} = \arg\max_{\eta} \sum_{\boldsymbol{S}} \sum_{\boldsymbol{K}} p(\boldsymbol{Y}, \boldsymbol{S}, \boldsymbol{K} \mid \eta, \phi_X)$$

η is the parameters of function which map the trained model to the model that match test data : $\phi_Y = G_\eta(\phi_X)$

Feature space adaptation

- Using EM algorithm to estimate $\hat{\upsilon}$
- E step : compute the auxiliary function:

$$Q(\upsilon,\upsilon') = E\{\log P(Y,S,K \mid \upsilon',\phi_X) \mid Y,\upsilon,\phi_X\}$$

$$= \sum_{S}\sum_{K} P(S,K \mid Y,\upsilon,\phi_X)\log P(Y,S,K \mid \upsilon',\phi_X)$$

- M step : maximizes the auxiliary function

$$\hat{\upsilon} = \arg\max_{\upsilon} Q(\upsilon,\upsilon')$$

Maximization step

$$Q(\upsilon, \upsilon') = \sum_{S}\sum_{K} P(\boldsymbol{S}, \boldsymbol{K} \mid \boldsymbol{Y}, \upsilon, \phi_X) \log \prod_{t=1}^{T} a_{S_t - 1, S_t} C_{S_t, k_t} P_y(\boldsymbol{y}_t \mid S_t, k_t, \upsilon', \phi_X)$$

$$P_y(\boldsymbol{y}_t \mid S_t, k_t, \upsilon', \phi_X) = \frac{Px(f_{\upsilon'}(\boldsymbol{y}_{t)} \mid S_t, k_t, \upsilon', \phi_X)}{\mid J_{\upsilon'}(\boldsymbol{y}_t) \mid}$$

$$J_{\upsilon'}(\boldsymbol{y}_t)_{i, j} = \frac{\partial y_{i, j}}{\partial f_{\upsilon', j}(\boldsymbol{y}_i)}$$

$$Q(\upsilon, \upsilon') = \sum_{S}\sum_{K} P(\boldsymbol{S}, \boldsymbol{K} \mid \boldsymbol{Y}, \upsilon, \phi_X) \log \prod_{t=1}^{T} a_{S_t - 1, S_t} C_{S_t, k_t} \frac{N(f_{\upsilon'}(\boldsymbol{y}_t); \boldsymbol{\nu}_{S_t, k_t}, \sum_{S_t, k_t}))}{\mid J_{\upsilon'}(\boldsymbol{y}_t) \mid}$$

Maximization step

$$
\begin{aligned}
Q(\upsilon,\upsilon') = & \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{t=1}^{T} P(S_{t-1}=i, S_t = j \mid \boldsymbol{Y}, \upsilon, \phi_X) \log a_{ij} \\
& + \sum_{j=1}^{N}\sum_{k=1}^{N}\sum_{t=1}^{T} P(S_t = j, c_t = k \mid \boldsymbol{Y}, \upsilon, \phi_X) \log c_{jk} \\
& + \sum_{j=1}^{N}\sum_{k=1}^{N}\sum_{t=1}^{T} P(S_t = j, c_t = k \mid \boldsymbol{Y}, \upsilon, \phi_X) \log N(f_{\upsilon'}(\boldsymbol{y}_t) : \mu_{jk}, \Sigma_{jk}) \\
& - \sum_{j=1}^{N}\sum_{k=1}^{N}\sum_{t=1}^{T} P(S_t = j, c_t = k \mid \boldsymbol{Y}, \upsilon, \phi_X) \log \mid J_{\upsilon'}(\boldsymbol{y}_t) \mid
\end{aligned}
$$

$$
\begin{aligned}
Q(\upsilon,\upsilon') = & \sum_{j=1}^{N}\sum_{k=1}^{N}\sum_{t=1}^{T} P(S_t = j, c_t = k \mid \boldsymbol{Y}, \upsilon, \phi_X) \\
& \times \{-\frac{1}{2}[f_{\upsilon'}(\boldsymbol{y}_t) - \mu_{jk}]^T \Sigma_{jk}^{-1}[f_{\upsilon'}(\boldsymbol{y}_t) - \mu_{jk}] - \log \mid J_{\upsilon'}(\boldsymbol{y}_t) \mid\}
\end{aligned}
$$

Maximization step

$$\boldsymbol{x}_t = f_v(\boldsymbol{y}_t) \Rightarrow \boldsymbol{c}' = f_w(\boldsymbol{c})$$

Add the weight constraint "$\boldsymbol{i}^T\boldsymbol{w} = K$" to the objective function

$$F(w, w') = Q(w, w') + \lambda(\boldsymbol{i}^T\boldsymbol{w} = K)$$

The maximum is obtained by differentiating it with respect to w' and solve for it's zero using Lagrange's multiplier λ

Model space adaptation

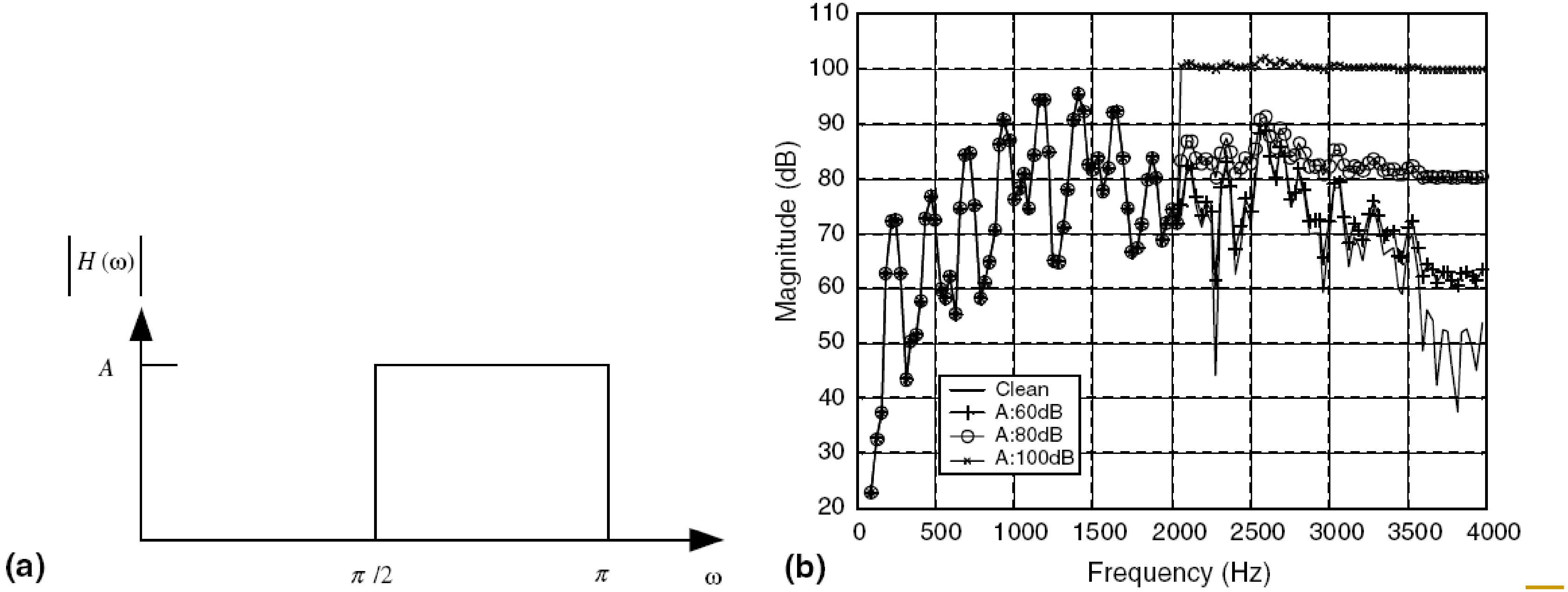
- Using EM algorithm similar to the feature space to estimate the weight for model space.

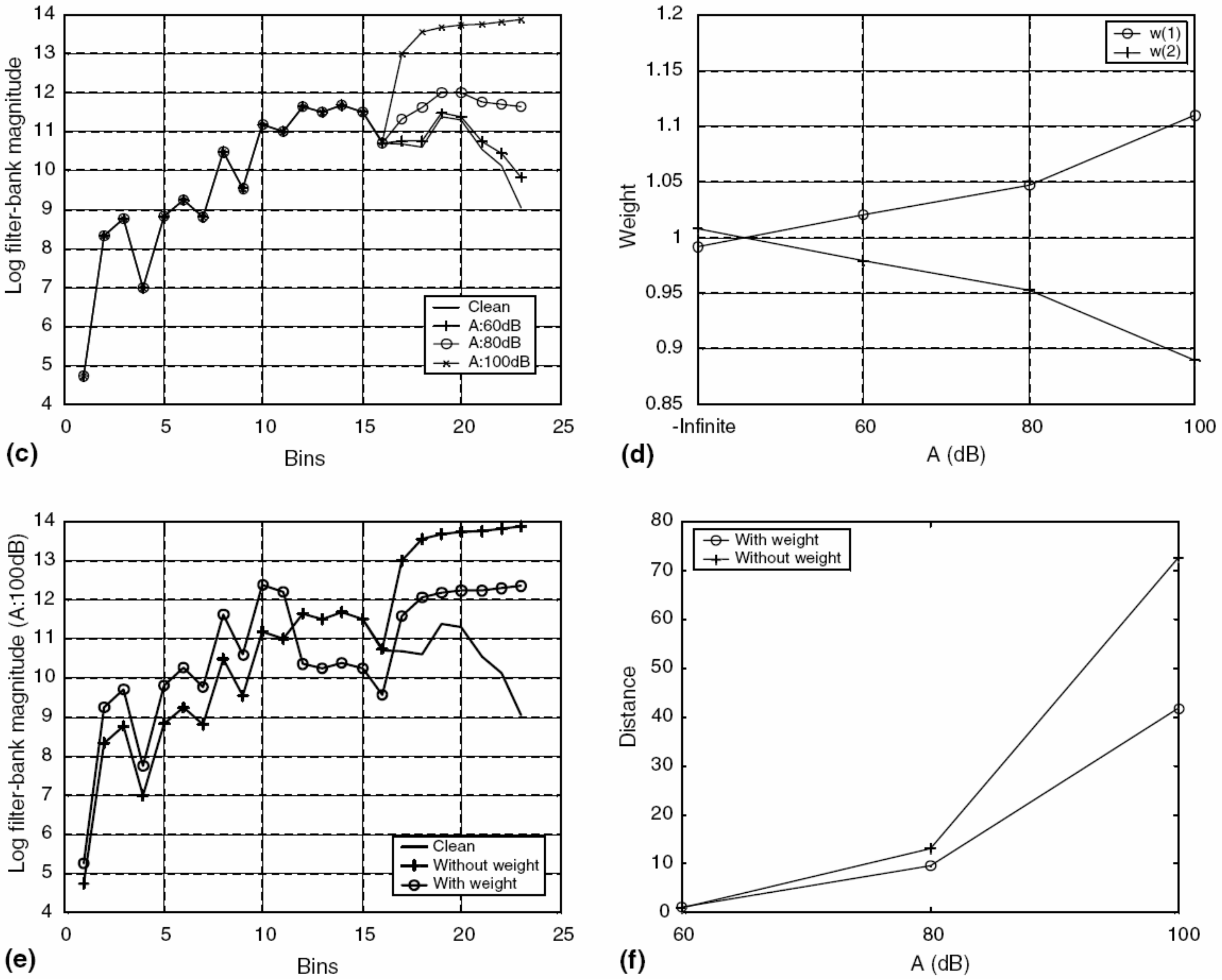
Option of model space adaptation

- Multiple class case
    - Separate the Gaussian components in the model set into multiple classes, and multiply each class with a different weight.
- Unconstrained weighting case
    - The weighting procedure on mean vectors may be regarded as a type of transformation on them. So the constraints may be removed.

Experiments on AURORA2

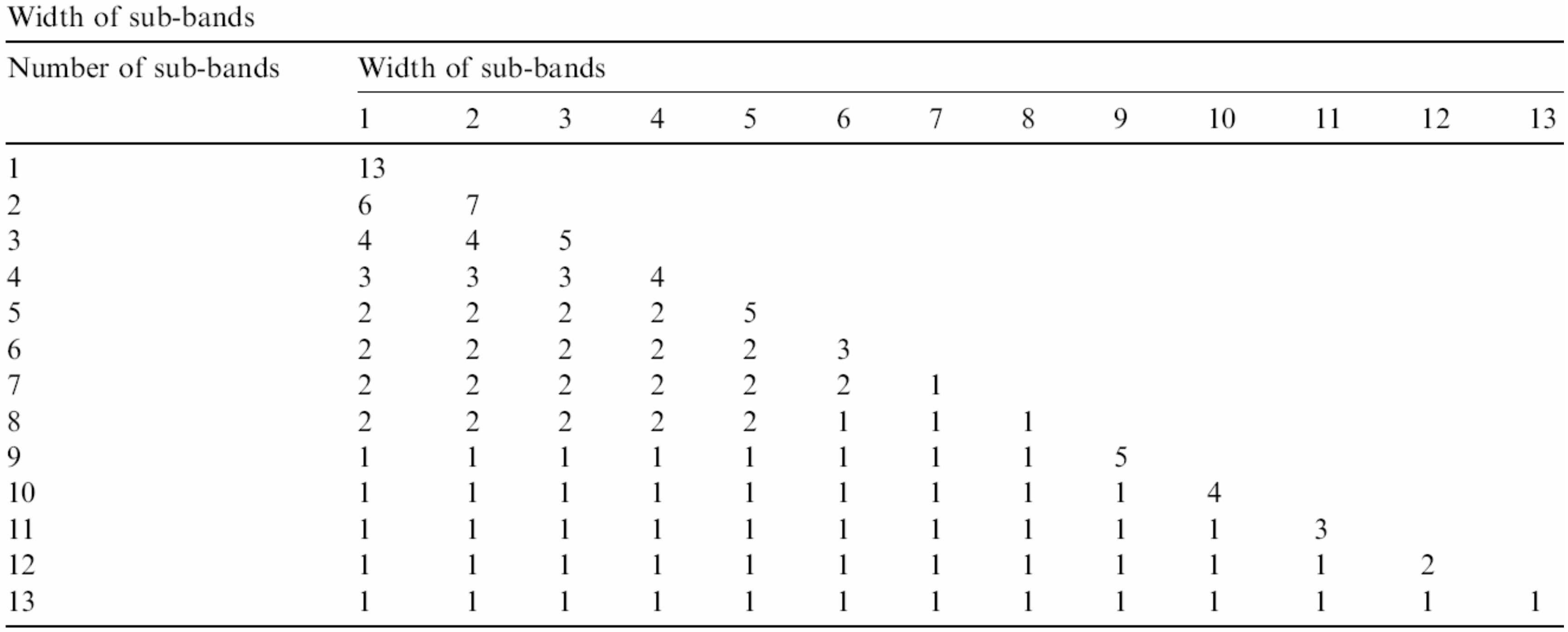
- A clean speech utterance was artificially supplied with noise signals that had special spectrums and used for adaptation.




Log filter-bank magnitude
14
13
12
11
10
9
8
7
6
5
4
0
5
10
15
20
25
Bins
(c)
Clean
A:60dB
A:80dB
A:100dB
Weight
1.2
1.15
1.1
1.05
1
0.95
0.9
0.85
-Infinite
60
80
100
A (dB)
(d)
w(1)
w(2)
Log filter-bank magnitude (A:100dB)
Bins
(e)
Clean
Without weight
With weight
Distance
80
70
60
50
40
30
20
10
0
60
80
100
A (dB)
(f)
With weight
Without weight

Width of sub-bands

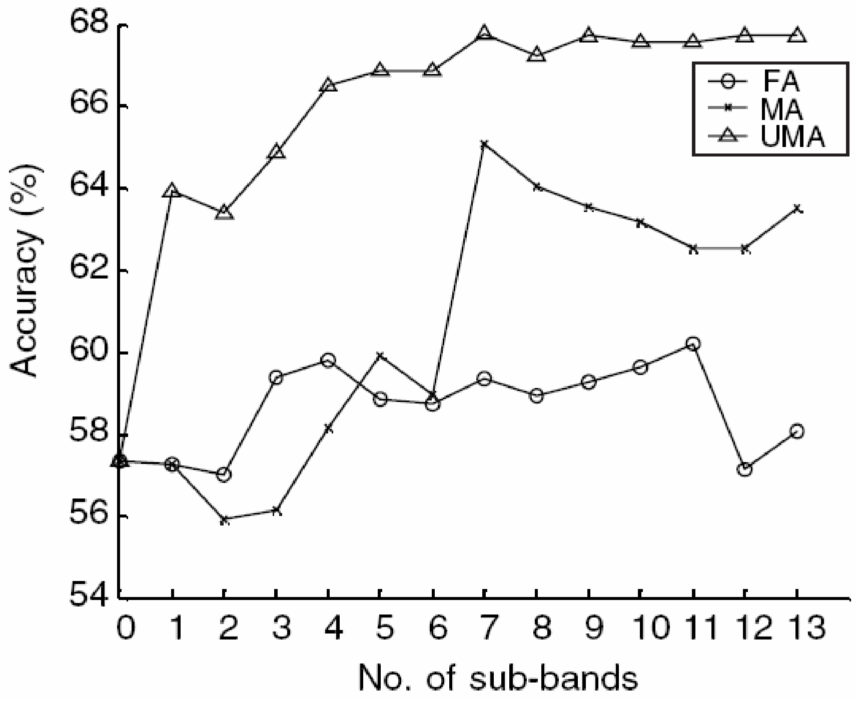
The log FBEs are divided into approximately uniform sub-bands.



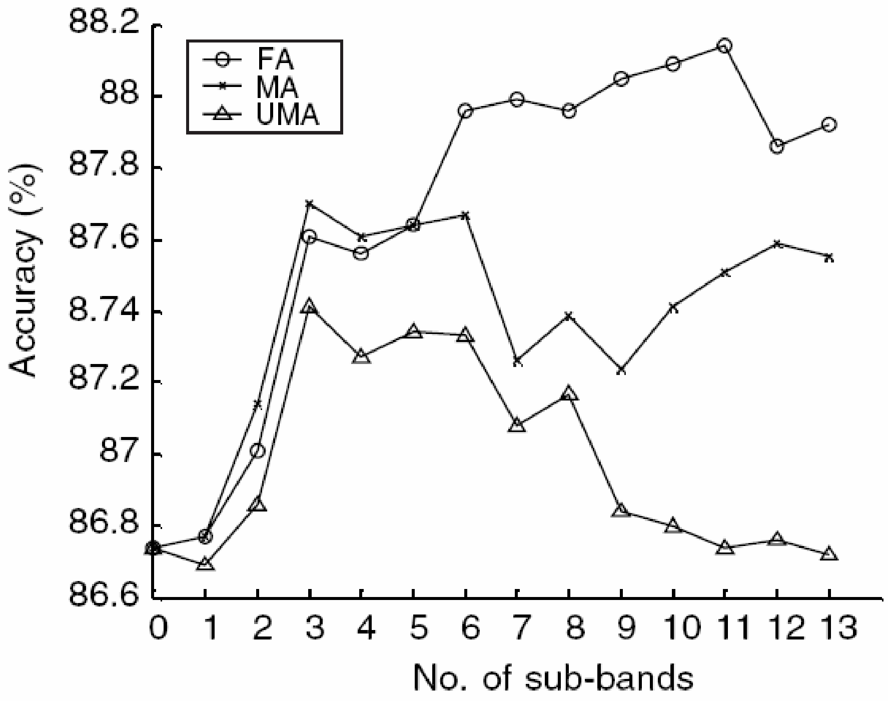
Width of sub-bands

Number of sub-bands	Width of sub-bands												
	1	2	3	4	5	6	7	8	9	10	11	12	13
1	13												
2	6	7											
3	4	4	5										
4	3	3	3	4									
5	2	2	2	2	5								
6	2	2	2	2	2	3							
7	2	2	2	2	2	2	1						
8	2	2	2	2	2	1	1	1					
9	1	1	1	1	1	1	1	1	5				
10	1	1	1	1	1	1	1	1	1	4			
11	1	1	1	1	1	1	1	1	1	1	3		
12	1	1	1	1	1	1	1	1	1	1	1	2	
13	1	1	1	1	1	1	1	1	1	1	1	1	1

Number of sub-bands

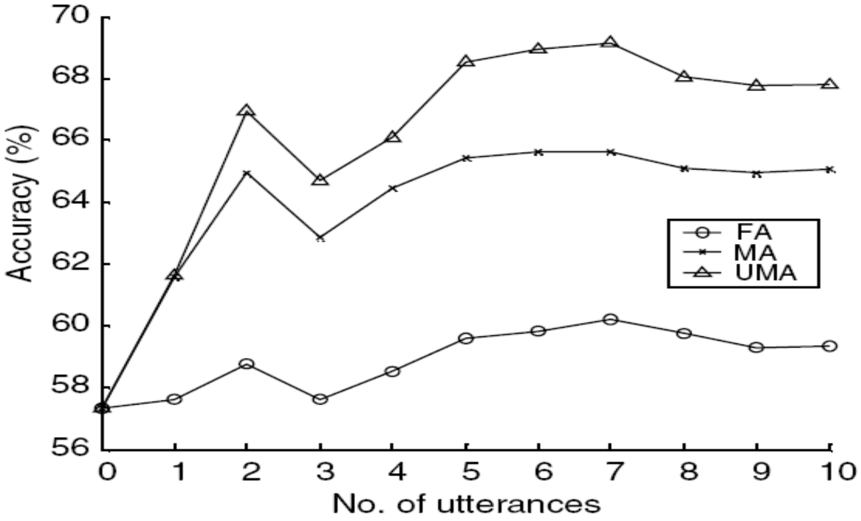


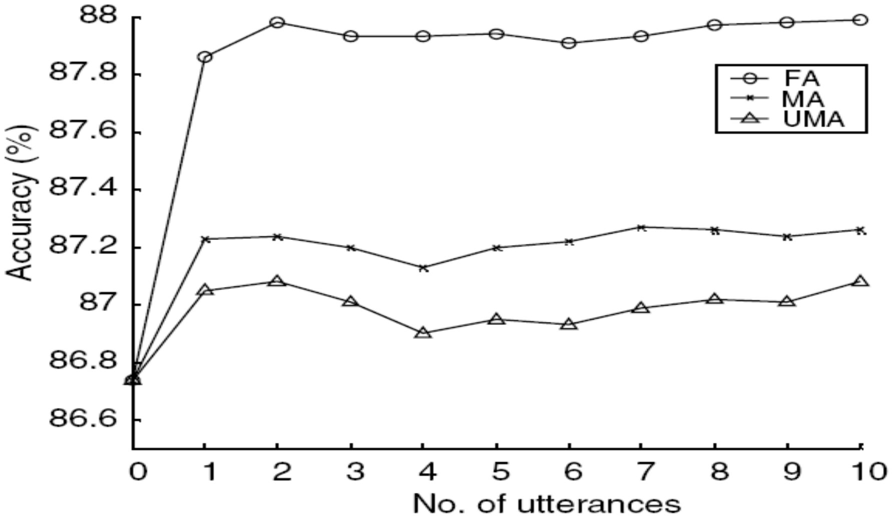
clean HMM



multi-condition HMM

Amount of adaptation data



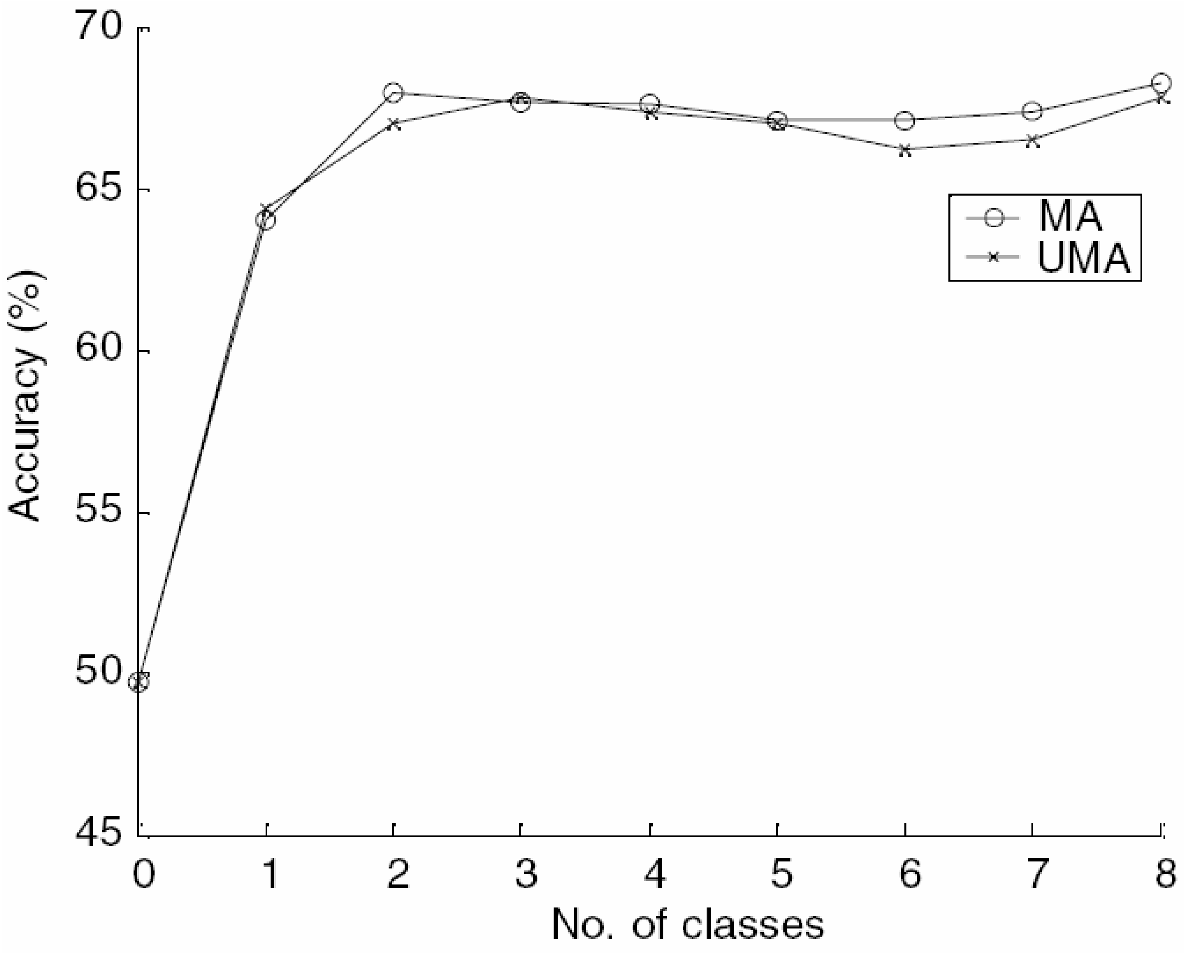


clean HMM

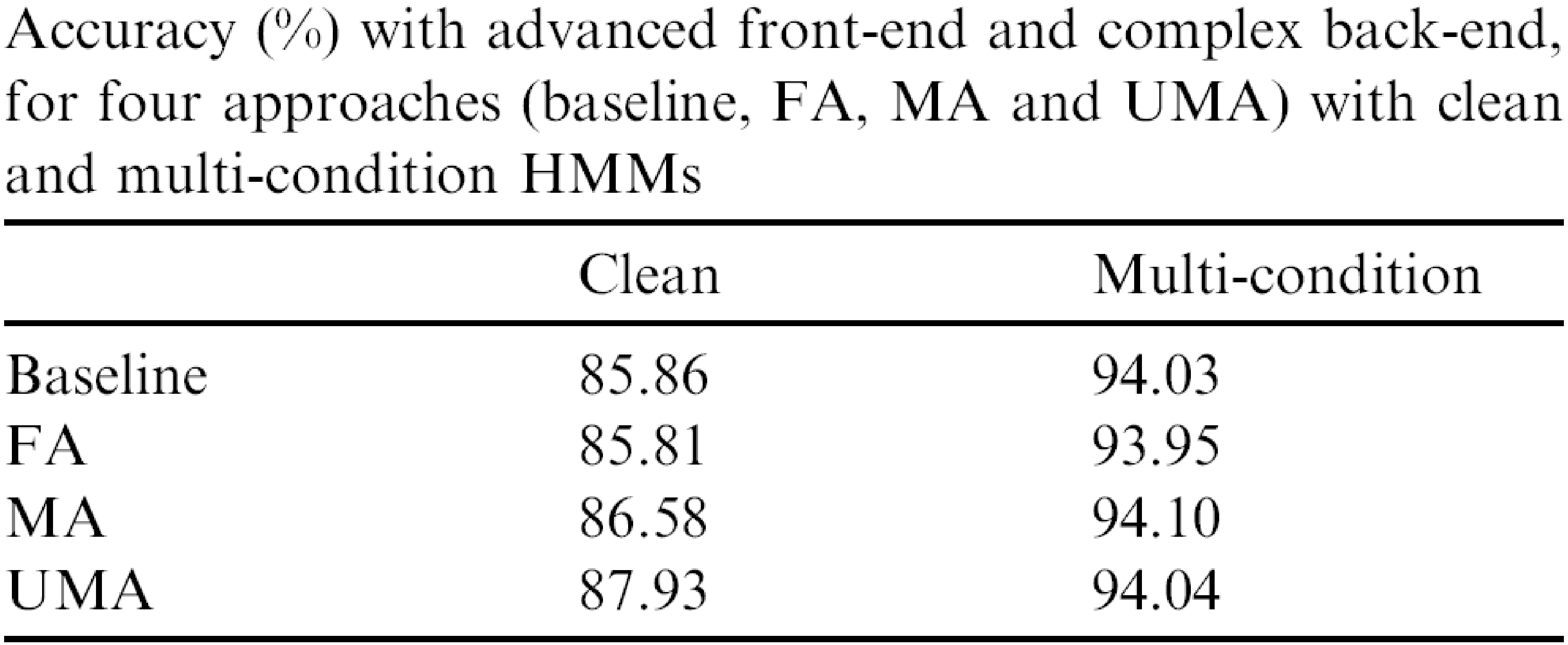
multi-condition HMM

- FA is more effective when the HMMs are less discriminative while MA and UMA are more effective when the HMMs are more discriminative.

Number of model classes



Results on advanced front-end



Accuracy (%) with advanced front-end and complex back-end, for four approaches (baseline, FA, MA and UMA) with clean and multi-condition HMMs

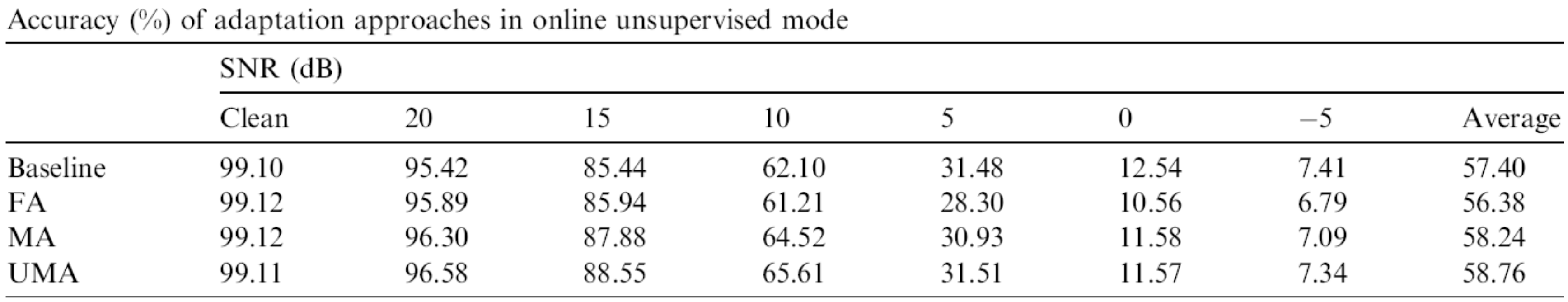
	Clean	Multi-condition
Baseline	85.86	94.03
FA	85.81	93.95
MA	86.58	94.10
UMA	87.93	94.04

Results of online unsupervised adaptation

- Noise type and level are unknown.

1. Recognize the utterance with baseline models.
2. Given the recognized transcription, adapt the weights in FA or models in MA and UMA.
3. Re-recognize the utterance with the adapted weights or models.

Results of online unsupervised adaptation



Accuracy (%) of adaptation approaches in online unsupervised mode

	SNR (dB)							
	Clean	20	15	10	5	0	–5	Average
Baseline	99.10	95.42	85.44	62.10	31.48	12.54	7.41	57.40
FA	99.12	95.89	85.94	61.21	28.30	10.56	6.79	56.38
MA	99.12	96.30	87.88	64.52	30.93	11.58	7.09	58.24
UMA	99.11	96.58	88.55	65.61	31.51	11.57	7.34	58.76

Conclusion

- This approach is useful when additive background noise signals were band-limited.