SPEECH RECOGNITION AND ENHANCEMENT USING SOME ROBUST HOS-BASED AR ESTIMATION TECHNIQUES

Josep M. SALAVEDRA*, Javier HERNANDO*, Enrique MASGRAU**, Asunción MORENO*

** Department of Signal Theory and Communications. Universitat Politècnica de Catalunya.Apartat. 30002. 08080-BARCELONA. SPAIN. Tel/Fax: +34-3- 4017404 / 4016447 . E-mail: mia@tsc.upc.es*

*** Department of Electrical Engineering and Computers. Universidad de Zaragoza. María de Luna, 3. 50015-ZARAGOZA. SPAIN*

ABSTRACT

We study some speech enhancement algorithms based on the iterative Wiener filtering method due to Lim-Oppenheim [2], where the AR spectral estimation of the speech is carried out using a second-order analysis. But in our algorithms we consider an AR estimation by means of cumulant analysis. This work extends some preceding papers due to the authors, where information of previous speech frames is taken to initiate speech AR modelling of the current frame. Two parameters are introduced to dessign Wiener filter at first iteration of this iterative algorithm. These parameters are the Interframe Factor (IF) and the Previous Frame Iteration (PFI). A detailed study of them shows they allow a very important noise suppression after processing only first iteration of this algorithm, without any appreciable increase of distortion. Finally, these cumulant-based algorithms are applied to Speech Recognition.

1. INTRODUCTION

It is well known, that many applications of speech processing that show very high perfomance in laboratory conditions degrade dramatically when working in real environments because of low robustness. The solution we propose here concerns to a preprocessing front-end in order to enhance the speech quality by means of a speech parametric modelling insensitive to the noise. The use of HO cumulants for speech AR modelling calculation provides the desirable uncoupling between noise and speech. It is based on the property that for Gaussian processes only, all cumulants of order greater than two are identically zero [1]. Moreover, the non-Gaussian processes presenting a symmetric p.d.f. have null odd-order cumulants. Considering a Gaussian or a symmetric p.d.f. noise (a good approximation of very real environments) and the non-Gaussian characteristic of the speech (principally for the voiced frames) it would be possible to obtain an spectral AR modelling of the speech more independent of the noise by using, e.g., third-order cumulants of noisy speech instead of common second-order statistics.

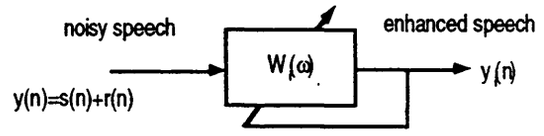
This work was supported by TIC 92-0800-C05-04

2. ITERATIVE WIENER ALGORITHM

In the original Lim-Oppenheim Method [2], noisy speech is enhanced by means of an iterative Wiener filtering that is defined as:

$$W(\omega) = \frac{Ps}{Ps + Pr} \qquad (1)$$

where Pr is the spectrum of the noise signal r(n), estimated in non-speech frames, and Ps is a spectrum estimation of the unavailable clean speech signal. So, both speech and noise spectra estimation must be available to design the Wiener filter at every frame. We talk over signal estimation because both signals are not available and only noisy speech signal can be processed. An iterative Wiener filtering is used to obtain a better estimation of the AR speech modelling (see Fig.1). At first sight an improvement of performance can be expected after every iteration since this current AR speech estimation is carried out from a cleaner speech signal than filter estimation of the preceding iteration. But other factors sidetrack this iterative algorithm and a limitation in the number of iterations must be taken in account. Clearly the filtered speech signal contains a smaller residual noise but it presents a larger spectral distortion. Therefore, increasing the number of iterations doesn't always involve a better speech estimation. It is well known that this algorithm leads to a narrowness and a shifting of the speech formants [3], providing an unnatural sounding speech. In [6] a detailed convergence analysis of this algorithm is carried out. It is proved that this estimated Wiener filter tends to cancel all signal frequencies with SNR lower than 4.77dB, and an additional attenuation, proporcionally to the noise level, affects signal frequencies with higher SNR, in comparison to the optimum Wiener filter. Only the non-contaminated speech frequencies undergo a null attenuation.



$$W_i(\omega) = \frac{P_{y_{i-1}}(\omega)}{P_{y_{i-1}}(\omega) + P_r(\omega)} \qquad \text{where} \quad P_{y_{i-1}}(\omega) = \frac{g^2}{\left|1 + \sum_{k=1}^{P} a_k e^{-jk\omega}\right|^2}$$

Figure 1. Scheme of the iterative Wiener algorithm.

3. THE PARAMETERIZED ALGORITHM

A parameterized Wiener filtering has been considered to have a better control over noise suppression, intelligibility loss and computational complexity, by adding two parameters ∂ and ß in the Wiener filter computation (1). So, we consider now the following equation:

$$W_i(w) = \left(\frac{P_y}{P_y + ß.P_r}\right)^{\partial} \qquad (2)$$

By varying these parameters ∂,ß, filters with different characteristics can be obtained. Thus, if ∂=ß=1 then expression (2) corresponds to the general Wiener filter equation (1), and if ∂=0.5, ß=1.0 it corresponds to power spectrum filtering. In [7], a detailed study of performance was reported. High values of both parameters lead to a more aggresive Wiener filter and so noise suppression is increased but distortion increases too. We found that ∂=1.0, ß=1.2 is a good trade-off among noise suppression, distortion, computational complexity and convergence speed of the iterative filtering, when 3rd-order statistics and low SNR are considered.

AR modelling of the speech spectrum estimation is obtained from third-order cumulants. Speech AR modelling coefficients a_k are computed by solving Third-Order Yule-Walker equations [4], [5]:

$$\sum_{k=0}^{p} a_k . C_3(i-k, j) = 0 \quad ,, 1 < i \le p \;\; ; \;\; -p \le j \le 0 \qquad (3)$$

where p=10 is the order of the AR filter. Fourth-order cumulant-based AR parameters are obtained by means of the same procedure, but we consider fourth-order cumulants instead of third-order ones and last dimension of fourth-order cumulants is frozen to zero lag. This procedure, that considers p+1 cumulant slices, presents a full-rank solution and it is unique [5].

As discussed in preceding works due to the authors [7], [8], we obtain a twofold benefit by considering this 3rd-order AR modelling: Firstly, an accelerated convergence of the iterative algorithm and so a reduction of both computational complexity and intelligibility loss; Secondly, achievement of a non polluted AR speech parameterization. In comparison to 2nd-order statistics estimation we obtain a good improvement but the price we pay for these advantages is a higher distortion. Thus a higher "peaking" or "narrowness" effect of the speech formants is brought about [6].

When additive noise is AWGN (SNR=0dB) the improvement over 2nd-order algorithm is very appreciated for any number of iterations (see Table.1). Thus, in Fig.2 an uniform improvement, iteration by iteration, is obtained when classical second-order statistics algorithm is evaluated. This improvement is similar when different values of Signal-to-Noise-Ratios are simulated. We may conclude that noise suppression saturates after 4 or 5 iterations of the iterative Wiener Algorithm, because other effects, such as intelligibility loss, overcome noise reduction. While the improvement of second-order approach increases gradually, but slowly, iteration by iteration, third-order one gets a very good improvement, about 3dB, after only two iterations and thus it obtains a faster convergence. In Fig.3, a lower noise sensitivity may be observed in the Third-Order Statistics domain: saturation effect appears after only 2 or 3 iterations in low SNR environments, and noise reduction effect is over-riding just in the first iteration when medium and high SNR environments are simulated.

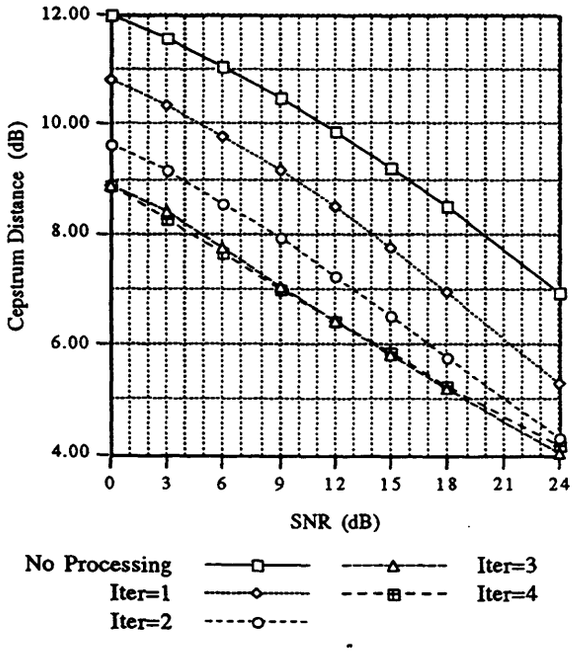


Figure 2. *Noise Suppression achieved by iterative Wiener filtering using classical autocorrelation function.*

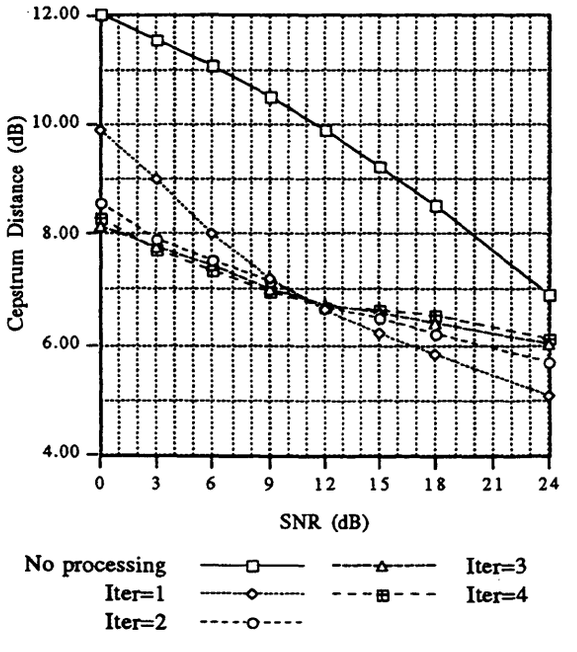


Figure 3. *Noise Suppression achieved by parameterized iterative Wiener filtering using third-order cumulants.*

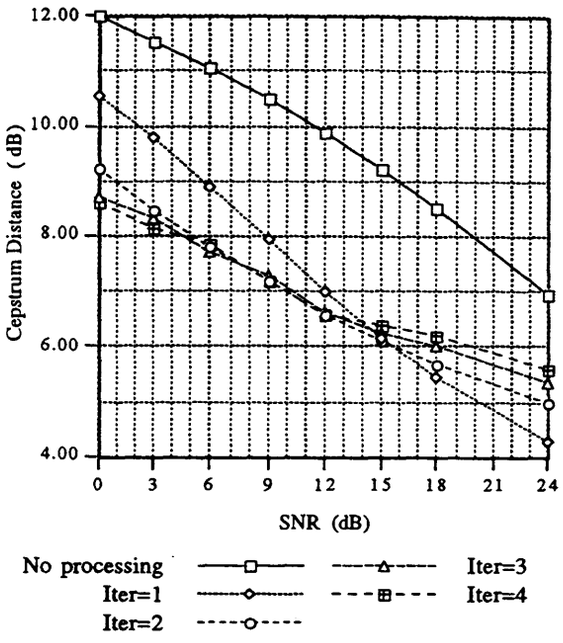


Figure 4. *Noise Suppression achieved by parameterized iterative Wiener filtering using fourth-order cumulants.*

Fig.4 shows performance of the fourth-order cumulant algorithm (AR4). It can be emphasized its lower aggressiveness, in comparison with third-order cumulant algorithm (AR3). Therefore, AR4 algorithm overcomes the other preceding algorithms (AR2 and AR3) when it works in quiet conditions (high SNR environments), where just first iteration must be processed. To conclude, AR3 algorithm seems to be the best choice in noisy conditions (low and medium SNR environments): it obtains better performance with respect to AR4 algorithm and, furthermore, its computational complexity is lower. Moreover, third-order cumulants lead to a faster noise reduction because of its higher aggressiveness with respect to both fourth-order cumulants and autocorrelation function.

4. THE INTERFRAME FACTOR IF

In table 1, we may appreciate an improvement that increases gradually iteration by iteration. Most part of noise reduction is obtained after processing two iterations. Third-order cumulants obtain an appreciable noise suppression (about 2dB in Cepstrum distance) after first iteration (see Table 1.b) and then this speech modelling is enhanced enough (Cepstrum distance decreases 3.5dB) in the second iteration because it estimates Wiener filter from a cleaner speech signal. At first iteration, speech AR modelling is computed from noisy signal without any initial information about the features of speech signal corresponding to the current frame. However, we know some information of the current speech frame by considering that speech signal features don't vary a lot between two consecutive overlapped frames. Therefore, we propose to obtain the first iteration AR coefficients as a combination between current frame AR estimation and previous frame AR coefficients. Thus, we dessign the non-causal Wiener filter, corresponding to first iteration, as a linear combination of coefficients a_k, belonging to two consecutive frames, calculated as follows:

$$A_k(n,1) = IF \cdot a_k(n,1) + (1 - IF) \cdot a_k(n-1,PFI) \qquad (4)$$
$$0 \le k \le P ; \quad 1 \le PFI \le MAXITER ; \quad 0 \le IF \le 1$$

where n is the current frame, PFI is the Previous Frame Iteration that we consider to help first iteration of the current frame and IF is the Interframe Factor. We write a_k when coefficients are estimated directly from a noisy speech frame and we note capital letter A_k when coefficients are coming from a linear combination of a_k . At the beginning of every speech activity we set parameter IF=1 because information of last speech frame is not related to the current speech frame. Wiener filter dessigns corresponding to the remaining iterations of the algorithm are estimated over a cleaner speech signal coming from Wiener filtering Output of previous iteration of the same frame:

$$A_k(n,iter) = a_k(n,iter) \quad ,, \quad 2 \le iter \le MAXITER \qquad (5)$$

where iter is the iteration number of the current frame. We have two parameters to control this linear combination. First parameter is the Interframe Factor IF that represents the amount of current speech AR estimation $a_k(n,1)$ we put in the AR modelling $A_k(n,1)$ of the filter. The interframe factor is the main parameter to control linear combination (4) because parameter IF=1 represents that only current AR estimation is considered to dessign Wiener filter at first iteration of current frame and then parameter PFI has no sense to be considered. Thus, parameter IF=1 refers to a situation where no interframe factor is defined. If we decide to consider previous frame information (IF<1) we must consider parameter PFI to answer the following question : Which iteration number (PFI) of preceding frame must we select to obtain a reliable speech AR modelling? Preceding works [7],[8] have shown that it is worthless to process more than 5 iterations while third-order statistics are considered. Therefore, parameter MAXITER=5 has been fixed in all our tests. On the other hand, parameter IF=0 represents that the coming noisy speech frame is filtered by means of a filter estimation coming from preceding speech frame.

Two different situations may be distinguished: PFI=1 and PFI>1. When information proceeding from first iteration of previous speech frame (PFI=1) is considered, no better results than before (IF=1) are expected, because the speech AR estimator is looking at the same noisy speech signal, but in the previous frame, and performance therefore decreases when parameter IF decreases to zero (see 1st iteration line in Fig.5). However, a good improvement (about 1.5dB in Cepstrum distance) is obtained when parameter IF=0 and PFI>1 but distortion effect increases more than 2dB in Cepstrum distance because current Wiener filter is dessigned with speech AR estimation proceeding from the preceding frame over a cleaner speech signal.

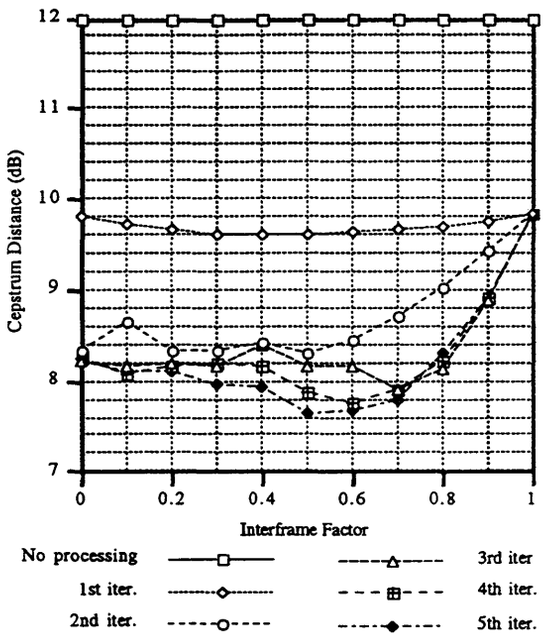


Figure 5. *Noise Suppression after processing first iteration of current frame when some different speech AR estimations belonging to different iterations of previous speech frame are considered.*

Clean speech has been processed by this system and so distortion effect corresponding to the iterative algorithm has been evaluated. To avoid an appreciable increase of distortion effect all values of parameter IF lower than 0.6 must be discarded. In Fig.5, first iteration of current frame corresponding to speech signal disturbed by AWGN at SNR=0dB has been processed and some different speech AR estimations of previous frame have been evaluated (ranging PFI from 1 to 5). We may come to the conclusion that values of parameter IF ranging from 0.6 to 0.8 represent a good trade-off between distortion and noise suppression. Therefore, we may achieve an improvement of 2dB in Cepstrum distance by introducing parameter IF (PFI=3 and IF=0.7) to estimate current speech AR modelling without any noticeable increase of distortion (0.25 dB). Thus, we may obtain an improvement higher than 4 dB in Cepstrum distance after processing only first iteration of the iterative Wiener filtering.

After second iteration most part of linear combinations leads to similar levels of Cepstrum Distance but, in listening tests, it may be appreciated a less distortion effect when parameter PFI>2 and furthermore the best performance is achieved after 3 iterations of Lim-Oppenheim algorithm while 4 iterations are necessary when parameter IF=1, to obtain a similar quality. Therefore, value PFI=3 may be considered as a good trade-off among computational complexity, distortion effect and noise suppression. This fact may be justified looking at Itakura Distance where a very important reduction is achieved (about 4.5dB), with only first iteration, and therefore formant estimation in voiced sounds is clearly improved by introducing these two parameters at first iteration of every frame. Obviously this linear combination of coefficients a_k tends to improve quality inside of voiced sounds. Some constraints have been added to the algorithm to protect unvoiced frames against this linear combination. Similar performance is achieved when diesel engine noise is considered, although differences between AR weighting method (IF<1) and no interframe weighting method (IF=1) are smaller.

5. SPEECH RECOGNITION EXPERIMENTS

This section reports the application of all parameterization techniques mentioned above to recognize isolated words in a speaker-independent task, with the HMM [10] approach, in order to compare their performance in the presence of additive white noise.

The database used in our experiments consists of ten repetitions of the Catalan digits uttered by seven male and three female speakers (1000 words) and recorded in a quiet room. Firstly, the system was trained with five of the speakers and tested with the others. Then the roles of both halves were changed and the reported results were obtained by averaging the two results.

The analog speech was first bandpass filtered to 100-3400 Hz. by an antialiasing filter, sampled at 8 KHz and quantized using two bytes per sample. The digitized clean speech was manually endpointed to determine the boundaries of each word. The endpoints obtained in this way were used in all our experiments including those in which noise was added to the signal. In this way we eliminate the effect of errors in endpoint detection on recognition accuracy and focus only on the recognition process itself. Clean speech was used for training in all the experiments. Noisy speech was simulated by adding zero mean white Gaussian noise to the clean signal so that the SNR of the resulting signal becomes 20, 10 and 0 dB. No preemphasis was performed.

In the parameterization stage of the recognition system, the signal was divided into frames of 30 ms. at a rate of 15 ms. and each frame was characterized by 10 cepstral parameters obtained either by the standard LPC method or the other techniques exposed in last section, using model order equal to 10. Obviously, these are not the optimum conditions for each parameterization technique but the results can help to compare their performance.

Before entering the recognition stage, the cepstral parameters were vector-quantized by means a codebook of 64 codewords using the standard Euclidean distance measure between cepstral vectors. This codebook size had been optimized in preliminary experiments using the standard LPC technique.

Each digit is characterized by a first order, left-to-right, discrete Markov model. The trade-off between computational load and recognition accuracy led us to consider models of 10

states without skips. Training and testing were performed using Baum-Welch and Viterbi algorithms, respectively [10].

The recognition rates obtained using the standard LPC technique were 58,7 %, 37,1 % and 24 %, for 20, 10 and 0 dB of SNR, respectively. However, using the new OSALPC representation [9], based on the LPC autocorrelation method applied on the one-sided autocorrelation sequence, the corresponding results were 88,3 %, 72,6 % and 35,8 %. As it can be seen, the OSALPC results are excellent and outperform considerably standard LPC ones in all noisy conditions tested. Regarding to the other HOS-based tecniques, their recognition results are between standard LPC rates and OSALPC rates. AR3 algorithm clearly overcomes AR2 one specially in noisy environments (SNR≤10dB).

6. CONCLUSIONS

A speech enhancement method based on an iterative Wiener filtering have been proposed. Spectral estimation of speech is obtained by means of an AR modelling based on cumulant analysis to provide the desirable noise-speech uncoupling. A comparison of different order cumulant-based algorithms is given. Two parameters, IF (Interframe Factor) and PFI (Previous Frame Iteration), have been considered to take advantage of previous speech spectrum estimations to initiate AR modelling corresponding to first iteration of the current speech frame. This approach achieves a noise suppression about 4dB (Cepstrum Distance) after processing only first iteration of the AR3 algorithm. This fact represents an improvement about 2dB (Cepstrum Distance) in relation to parameterized third-order algorithm (IF=1). Finally, the convergence of the iterative algorithms based on cumulant AR estimation is strongly accelerated. Therefore, a good reduction of computational complexity and processing delay is achieved, while no appreciable increase of distortion effect is generated. Furthermore, These cumulant-based algorithms have been integrated in a Speech Recognition System and some improvements have been reported. All these features are specially esteemed when low and medium SNR are considered.

7. REFERENCES

[1] **C.L.Nikias, M.R.Raghuveer,** "Bispectrum Estimation: A Digital Signal Processing Framework". Proc. of IEEE, pp. 869-891. July 1987.

[2] **J.S.Lim,A.V.Oppenheim,**"All-Pole Modeling of Degraded Speech".IEEE Trans ASSP, pp.197-210. June 1978.

[3] **J.H.L.Hansen, M.A.Clements,** "Constrained Iterative Speech Enhancement with Apllications to Spech Recognition". IEEE Trans ASSP, pp.795-805. April 1991.

[4] **A.Swami, J.M.Mendel,** "AR Identifiability using Cumulants". Proc. Workshop on HO Spectral Analysis, pp.13-18. Vail, CO, USA. June 1989.

[5] **G.B.Giannakis,** "On the Identifiability of non-Gaussian ARMA Models using Cumulants". IEEE Trans ASSP, pp.1284-1296. July 1990.

[6] **E.Masgrau, J.M.Salavedra, A.Moreno, A. Ardanuy,** "Speech Enhancement by Adaptive Wiener Filtering based on Cumulant AR Modelling". Proc. ESCA Workshop on Speech Processing in Adverse Conditions, pp 143-146. Cannes, France. November 1992.

[7] **J.M.Salavedra, E.Masgrau,A.Moreno, X.Jové and J.Estarellas,** "Robust Coefficients of a Higher-order AR Modelling in a Speech Enhancement System using parameterized Wiener Filtering". Proc. MELECON'94, pp. 69-72. Antalya, Turkey. April 1994.

[8] **J.M.Salavedra, E.Masgrau, A.Moreno, J.Estarellas,** "Some robust Speech Enhancement Techniques using Higher-order AR Estimation". Proc. EUSIPCO, pp.1194-1197. Edinburgh, Scotland. September 1994.

[9] **J.M.Salavedra, E.Masgrau, A.Moreno, J.Estarellas,** "Some fast Higher-order AR Estimation Techniques applied to parametric Wiener Filtering". Proc. ICSLP, pp.1655-1658. Yokohama, Japan. September 1994.

[10] **L.R. Rabiner,** "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition". Proc. IEEE, vol. 77, n.2. February 1989.

Table 1. *Distance measures using algorithms based on: a) second order statistics; b) parameterized third order cumulants; c) parameterized third order with interframe factor IF=0.7, considering parameter PFI=3; d) fourth order cumulants at SNR=0dB.*

a)	SNR	SEGSN	ITAKU	COSH	CEPST
0 iter.	0.00	0.79	9.57	11.67	12.02
1 iter.	7.36	4.38	9.21	10.71	11.01
2 iter.	8.83	5.92	8.86	10.17	9.90
3 iter.	9.04	6.16	7.30	9.04	9.34
4 iter.	9.11	6.25	6.42	8.45	9.20

b)	SNR	SEGSN	ITAKU	COSH	CEPST
0 iter.	0.00	0.79	9.57	11.67	12.02
1 iter.	7.92	4.86	8.18	9.78	9.82
2 iter.	7.60	5.31	5.94	8.16	8.47
3 iter.	7.59	5.59	5.11	7.55	8.15
4 iter.	7.36	5.79	5.15	7.64	8.30

c)	SNR	SEGSN	ITAKU	COSH	CEPST
0 iter.	0.00	0.79	9.57	11.67	12.02
1 iter.	8.31	5.17	5.00	7.82	7.90
2 iter.	7.90	5.56	5.03	7.54	8.08
3 iter.	7.91	5.88	5.01	7.43	7.98
4 iter.	7.68	5.78	4.86	7.45	8.16

d)	SNR	SEGSN	ITAKU	COSH	CEPST
0 iter.	0.00	0.79	9.57	11.67	12.02
1 iter.	7.47	4.53	8.97	10.49	10.53
2 iter.	7.39	4.95	7.88	9.65	9.30
3 iter.	7.37	5.11	6.55	8.65	8.80
4 iter.	7.77	5.49	5.52	7.91	8.47