SPEECH ENHANCEMENT USING A ROBUST KALMAN FILTER POST-PROCESSOR IN THE MODULATION DOMAIN

Yu Wang and Mike Brookes

Department of Electrical and Electronic Engineering,
Exhibition Road, Imperial College London, UK
Email: {yw09, mike.brookes}@imperial.ac.uk

ABSTRACT

We propose a speech enhancement algorithm that applies a Kalman filter in the modulation domain to the output of a conventional enhancer operating in the time-frequency domain. The speech model required by the Kalman filter is obtained by performing linear predictive analysis in each frequency bin of the modulation domain signal. We show, however, that the corresponding speech synthesis filter can have a very high gain at low frequencies and may approach instability. To improve the stability of the synthesis filter, we propose two alternative methods of limiting its low frequency gain. We evaluate the performance of the speech enhancement algorithm on the core TIMIT test set and demonstrate that it gives consistent performance improvements over the baseline enhancer.

Index Terms— speech enhancement, post-processing, Kalman filter, robust linear prediction, modulation domain

1. INTRODUCTION

The goal of a speech enhancement algorithm is to reduce or eliminate background noise without distorting the speech signal. Numerous speech enhancement algorithms have been proposed in the literature; among the most popular are those that apply a variable gain in the time-frequency domain such as the minimum mean square (MMSE) spectral amplitude [1] and log spectral amplitude [2] enhancers. These enhancement algorithms give dramatic improvements in signal-to-noise ratio (SNR) but at the expense of introducing spurious tonal artefacts known as musical noise and speech distortion. A number of authors have suggested removing the musical noise by applying some form of post-processing to the to output of the baseline enhancer or to the time-frequency gain function that it utilizes. Smoothing the enhancer gain function is used in [3] to attenuate musical noise in time frames with low SNR and in [4] the gain function of each frame is first transformed into the cepstral domain so that smoothing may be selectively applied to the high quefrency coefficients. In [5], median filtering is applied to time-frequency cells that are classified as having a low probability of containing speech energy in order to eliminate the isolated peaks that characterise musical noise.

Several authors have proposed speech enhancers that apply a Kalman filter (KF) to the time domain signal [6, 7, 8, 9] and more recently, So and Paliwal have proposed applying the KF to the short-time modulation domain instead [10]. In this paper, we propose the use of a KF in the modulation domain as a post-processor for speech that has been enhanced by an MMSE spectral amplitude algorithm [1]. The KF incorporates an autoregressive model for the time-evolution of the spectral amplitude in each frequency bin; this is estimated using linear predictive (LPC) analysis applied to the time-frequency domain output of the MMSE enhancer. Because the spectral amplitudes include a strong DC component, the gain of the corresponding LPC synthesis filter can be very high at low frequencies and we therefore propose two alternative ways of constraining the low frequency gain in order to improve the filter stability. The remainder of the paper is organized as following; in Section 2 we describe the KF technique for speech enhancement in the modulation domain and after that, in Section 3 we introduce the derivation of the two robust linear prediction models. Finally the evaluation of the new algorithms and the conclusions are given in Section 4 and 5, respectively.

2. MODULATION DOMAIN KALMAN FILTERING

Representing the amplitude spectrum of the noisy speech signal and the clean speech as $Y(n,k)$ and $S(n,k)$, respectively, we assume an additive model of the noisy speech as

$$Y(n,k) = S(n,k) + N(n,k) \tag{1}$$

where n denotes the acoustic frame and k denotes the acoustic frequency. To perform Kalman filtering in the modulation domain, each frequency bin is processed independently; for clarity, we omit the frequency index, k, in the description that follows.

We assume that the temporal envelope, $S(n)$, of the amplitude spectrum of speech signal can be modeled by a linear predictor with coefficients a_i $(1 \leq i \leq p)$ in each modulation frame:

$$S(n) = -\sum_{i=1}^{p} a_i S(n-i) + P(n) \tag{2}$$

where $P(n)$ is a random Gaussian excitation signal with variance σ_P^2. The equations for Kalman filtering in the modulation domain are given in detail in [10] and we give only a brief overview here. In the modulation domain, time-domain noise has colored characteristics [10] and hence a KF for removing a colored noise is used [6]. Within each frequency bin, we use autoregressive models for the speech and the noise of orders p and q respectively and so the state vector in our KF has dimension $p + q$. The state space representation is given by

$$\begin{aligned}\begin{bmatrix}\mathbf{S}(n)\\ \mathbf{N}(n)\end{bmatrix} &= \begin{bmatrix}\mathbf{A}(n) & \mathbf{0}\\ \mathbf{0} & \mathbf{B}(n)\end{bmatrix}\begin{bmatrix}\mathbf{S}(n-1)\\ \mathbf{N}(n-1)\end{bmatrix}\\ &\quad + \begin{bmatrix}\mathbf{d}_p & \mathbf{0}\\ \mathbf{0} & \mathbf{d}_q\end{bmatrix}\begin{bmatrix}P(n)\\ Q(n)\end{bmatrix}\end{aligned} \tag{3}$$

$$Y(n) = \begin{bmatrix}\mathbf{d}_p^T & \mathbf{d}_q^T\end{bmatrix}\begin{bmatrix}\mathbf{S}(n)\\ \mathbf{N}(n)\end{bmatrix} \tag{4}$$

where $\mathbf{S}(n) = [S(n) \cdots S(n-p+1)]^T$ is the speech state vector. $\mathbf{d}_p = [1\, 0 \cdots 0]^T$ is a p-dimensional vector and the speech transition matrix has the form $\mathbf{A}(n) = \begin{bmatrix} & -\mathbf{a}^T\\ \mathbf{I} & \mathbf{0}\end{bmatrix}$ where $\mathbf{a} = [a_1 \cdots a_p]^T$ is the LPC coefficient vector, and $\mathbf{0}$ denotes an all-zero column vector of length $p-1$. The quantities $\mathbf{d}_q$, $\mathbf{N}(n)$ and $\mathbf{B}(n)$ are defined similarly for the order-q noise model. The speech signal $S(n)$ is thus generated in the modulation domain as the output of the LPC synthesis filter defined as

$$H(z) = \frac{1}{1 + \sum_{i=1}^{p} a_i z^{-i}} \tag{5}$$

with the excitation signal $P(n)$.

To determine the speech and noise model parameters, the time-frequency signal is segmented into overlapping modulation frames. For each frequency bin, a speech model $\{\mathbf{a},\ \sigma_p^2\}$ is estimated by applying autocorrelation LPC analysis to the modulation frame. A separate voice activity detector is applied to each frequency bin and a noise model, $\{\mathbf{b},\ \sigma_q^2\}$, estimated during intervals where speech is absent. Full details are given in [10].

3. KALMAN FILTER POST-PROCESSING

The framework for our proposed speech enhancer is shown in Fig. 1 and differs from that in [10] in two respects which we have found to result in enhanced speech of improved quality. First, we apply the KF not to the spectrum of the original noisy speech signal but rather to that of the output of an enhancer that implements the spectral amplitude MMSE algorithm from [1]. Second, motivated by [11] and [12] we apply the KF to the cube-root of the short-time power spectrum rather than to the amplitude spectrum. Referring to Fig. 1, a short-time Fourier transform (STFT) is applied to the MMSE-enhanced speech and the cube-root of the resulting power spectrum is taken. In our baseline system, denoted KFMD in Sec. 4, the speech and noise models are estimated using the method of [10] and are used in the KF described in Sec. 2. The output from the KF is converted back to the amplitude domain, combined with the noisy phase spectrum and passed through an inverse-STFT to create the output speech. Although we do not do so in our implementation, it would be possible to eliminate the initial STFT operation by taking the MMSE enhancer output directly in the time-frequency domain.

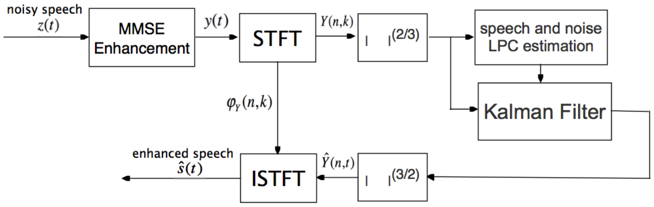

Fig. 1. Block diagram of KFMD algorithm

LPC is conventionally applied to a zero-mean time-domain signal but in the modulation domain KF, it is applied to a positive-valued sequence of transformed spectral amplitudes. As we will show, when LPC analysis is applied to a signal that includes a strong DC component, the resultant synthesis filter can have a very high gain at low frequencies and the filter may, as a consequence, be close to instability. We have found that this near-instability significantly degrades the quality of the output speech and thus in Sec. 3.2 and 3.3 we propose two alternative ways of preventing it.

3.1. Effect of DC bias on LPC analysis

In this section, we determine the effect of a strong DC component on the results of LPC analysis. Suppose first that $S(n)$ has zero mean and that the LPC coefficient vector, $\mathbf{a}$, for a frame of length N is determined from the Yule-Walker equations

$$\mathbf{a} = -\mathbf{R}^{-1}\mathbf{g} \tag{6}$$

where the elements of the autocorrelation matrix, $\mathbf{R}$, are given by $R_{i,j} = \frac{1}{N}\sum_n S(n-i)S(n-j)$ for $1 \le i, j \le p$ and the elements of $\mathbf{g}$ are $g_i = R_{i,0}$. The DC gain of the synthesis filter $H(z)$ in equation (5) is given by

$$G = \frac{1}{1 + \mathbf{w}^T\mathbf{a}}$$

where $\mathbf{w} = [1\, 1 \cdots 1]^T$ is a p-dimensional vector of ones.

If now a DC component, d, is added to each $S(n)$, the effect is to add d^2 onto each $R_{i,j}$ and the new LPC coefficients, $\mathbf{a}'$, are given by

$$\begin{aligned}\mathbf{a}' &= -\left(\mathbf{R}+d^2\mathbf{w}\mathbf{w}^T\right)^{-1}\left(\mathbf{g}+d^2\mathbf{w}\right)\\ &= -\left(\mathbf{R}^{-1}-\frac{d^2\mathbf{R}^{-1}\mathbf{w}\mathbf{w}^T\mathbf{R}^{-1}}{1+d^2\mathbf{w}^T\mathbf{R}^{-1}\mathbf{w}}\right)\left(\mathbf{g}+d^2\mathbf{w}\right)\end{aligned}$$

where the second line follows from the Matrix Inversion Lemma [13]. Writing $r = d^2\mathbf{w}^T\mathbf{R}^{-1}\mathbf{w}$, we can obtain

$$\mathbf{w}^T\mathbf{a}' = \frac{-\mathbf{w}^T\mathbf{R}^{-1}\mathbf{g}-r}{1+r} = \frac{\mathbf{w}^T\mathbf{a}-r}{1+r}$$

Thus the DC gain of the new synthesis filter is

$$\frac{1}{1+\mathbf{w}^T\mathbf{a}'} = \frac{1+r}{1+\mathbf{w}^T\mathbf{a}} \tag{7}$$

From (7) we see that the DC gain of the synthesis filter has been multiplied by $1+r$ where r is proportional to the power ratio of the DC and AC components of $S(n)$. If this ratio is large, the low frequency gain of the LPC synthesis filter can become very high which results in near instability and poor prediction. Accordingly, in the following sections we propose two alternative methods of limiting the low frequency gain of the LPC synthesis filter.

3.2. Method 1: Bandwidth Expansion

The technique of bandwidth expansion is widely used in coding algorithms to reduce the peak gain and improve the stability of an LPC synthesis filter [14]. If a modified set of LPC coefficient is defined by $\bar{a}_i = \alpha^i a_i$, for some constant $\alpha < 1$, then the poles of the synthesis filter are all multiplied by α. This moves the poles away from the unit circle thereby reducing the gain of the corresponding frequency domain peaks and improving the stability of the filter. In Sec. 4 we evaluate the effect of using this revised set of LPC coefficients, $\bar{\mathbf{a}}$, in the KF of Fig. 1 (denoted the "BKFMD" algorithm) and find that it results in a consistent improvement in performance.

3.3. Method 2: Constrained DC gain

Although the bandwidth expansion approach is effective in limiting the low frequency gain of the synthesis filter, it also modifies the filter response at higher frequencies thereby destroying its optimality. An alternative approach is to constrain the DC gain of the synthesis filter to a predetermined value and determine the optimum LPC coefficients subject to this constraint. As noted in Sec. 3.1, the DC gain of the LPC synthesis filter is given by G and we can force $G = G_0$ by imposing the constraint

$$\mathbf{w}^T\mathbf{a} = \frac{1-G_0}{G_0} \triangleq \beta > -1.$$

The average prediction error energy in the analysis frame is given by

$$E = \frac{1}{N}\sum_n\left\{S(n)+\sum_{i=1}^{p}a_iS(n-i)\right\}^2$$

and we would like to minimize E subject to the constraint $\mathbf{w}^T\mathbf{a} = \beta$. Using a Lagrange multiplier, λ, the solution, $\tilde{\mathbf{a}}$ to this constrained optimization problem is obtained by solving the $p+1$ equations

$$\begin{aligned}\frac{d}{da_i}\left(E+\lambda\mathbf{w}^T\tilde{\mathbf{a}}\right) &= 0\\ \mathbf{w}^T\tilde{\mathbf{a}} &= \beta\end{aligned}$$

and the solution is

$$\begin{pmatrix}0.5\lambda\\ \tilde{\mathbf{a}}\end{pmatrix} = \begin{pmatrix}0 & \mathbf{w}^T\\ \mathbf{w} & \mathbf{R}\end{pmatrix}^{-1}\begin{pmatrix}\beta\\ -\mathbf{g}\end{pmatrix} \tag{8}$$

where $\mathbf{R}$, $\mathbf{g}$ and $\mathbf{w}$ are as defined in Sec. 3.1. In Sec. 4 we evaluate the effect of using this revised set of LPC coefficients, $\tilde{\mathbf{a}}$, in the KF of Fig. 1 (denoted the "CKFMD" algorithm) and find that it results in a consistent improvement in performance both over the KFMD algorithm, which uses the unconstrained filter coefficients, and also over the BKFMD algorithm which uses the bandwidth expanded coefficients.

4. IMPLEMENTATION AND EVALUATION

4.1. Stimuli of experiments

In this section, we compare the performance of the baseline MMSE enhancer [15] with that of the three algorithms that incorporate a KF postprocessor. The KFMD algorithm uses an unconstrained speech model, the BKFMD algorithm incorporates the bandwidth expansion from Sec. 3.2 while the CKFMD algorithm uses the constrained filter from Sec. 3.3. In our experiments, we use the core test set from the TIMIT database [16] which contains 16 male and 8 female speakers each reading 8 distinct sentences (totalling 192 sentences) and the speech is corrupted by white and factory noise from the RSG-10 database [17] at -5, 0, 5, 10, 15, and 20 dB signal-to-noise ratio (SNR). The algorithm parameters were determined by optimizing performance on a subset of the TIMIT training set. We use an acoustic frame length of 32 ms with a 4 ms frame increment which gives a sample rate of 250 Hz in the modulation domain. The speech model is determined from a modulation frame of 128 ms (32 acoustic frames) with a 16 ms frame increment. For the KFMD algorithm, the speech and noise models are of orders $p = 2$ and $q = 4$ respectively while for the BKFMD and CKFMD algorithms, they are $p = 3$ and $q = 6$, as the different p and q give the best performance for the corresponding enhancers. Additionally, we set $\alpha = 0.7$ and $\beta = -0.8$ and use a Bartelett-Hanning window in the analysis-synthesis procedure and a Hamming window for the estimation of the speech model coefficients.

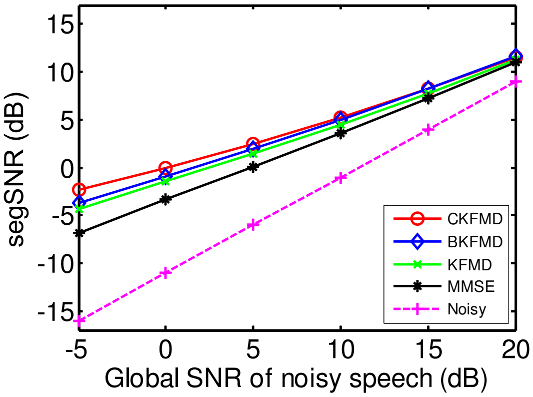

Fig. 2. Average segSNR values comparing different algorithms, where speech signals are corrupted by white noise at different SNR levels.

4.2. Performance of new algorithms

Using the new LPC models, the performance of the speech enhancers is evaluated using both segmental SNR (segSNR) and the perceptual evaluation of speech quality (PESQ) measure defined in ITU-T P.862. In all cases the measures are averaged over the 192 sentences in the TIMIT core test set. Figures 2 and 3 show how the average segSNR varies with global SNR for white noise and factory noise for the unenhanced speech, the baseline MMSE enhancer and the three KF postprocessing algorithms presented here. We see that at high SNRs, all the algorithms have very similar performance. However at 0 dB SNR the KFMD provides an approximate 2 dB improvement in segSNR over MMSE enhancement and the BKFMD and CKFMD algorithms give an additional 0.5 and 1.5 dB improvement respectively. The PESQ results shown in Fig. 4 and 5 broadly mirror the segSNR results although the post-processing gives an improvement in PESQ even at high SNRs. For both noise types, the constrained KF postprocessor (CKFMD) gives a PESQ improvement of >0.2 over a wide range of SNRs. In addition, informal listening tests also indicate that the proposed post-processing methods, especially BKFMD and CKFMD enhancers, are able to reduce the musical noise introduced by MMSE enhancer.

5. CONCLUSION

We have proposed three alternative methods of post-processing the output of an MMSE spectral amplitude speech enhancer by using a KF in the modulation domain. The three methods differ in how they estimate the LPC speech model in each modulation frame. We have shown that all three methods give consistent improvements over the MMSE enhancer in both segSNR and PESQ and that the best method, which performs LPC analysis with a constrained DC gain, improves PESQ scores by at least 0.2 over a wide range of SNRs.

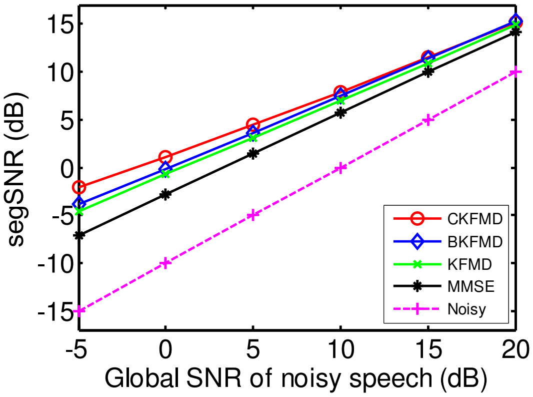

Fig. 3. Average segSNR values comparing different algorithms, where speech signals are corrupted by factory noise at different SNR levels.

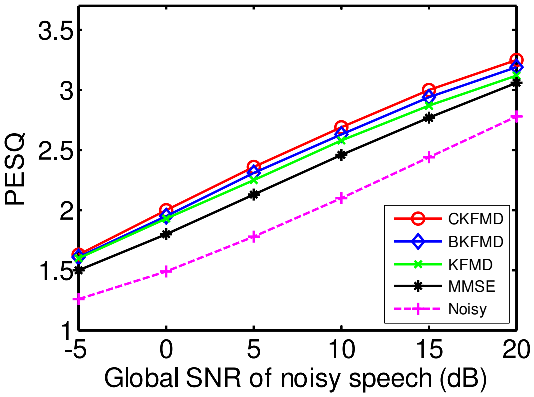

Fig. 4. Average PESQ values comparing different algorithms, where speech signals are corrupted by white noise at different SNR levels.

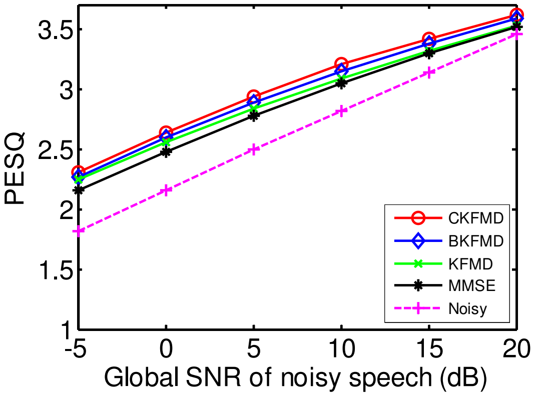

Fig. 5. Average PESQ values comparing different algorithms, where speech signals are corrupted by factory noise at different SNR levels.

6. REFERENCES

[1] Y. Ephraim and D. Malah. Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator. *IEEE Trans. Acoust., Speech, Signal Process.*, 32(6):1109–1121, December 1984.

[2] Y. Ephraim and D. Malah. Speech enhancement using a minimum mean-square error log-spectral amplitude estimator. *IEEE Trans. Acoust., Speech, Signal Process.*, 33(2):443–445, 1985.

[3] T. Esch and P. Vary. Efficient musical noise suppression for speech enhancement system. In *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4409–4412, April 2009.

[4] C. Breithaupt, T. Gerkmann, and R. Martin. Cepstral smoothing of spectral filter gains for speech enhancement without musical noise. *Signal Processing Letters, IEEE*, 14(12):1036–1039, December 2007.

[5] Zenton Goh, Kah-Chye Tan, and T. G. Tan. Postprocessing method for suppressing musical noise generated by spectral subtraction. *IEEE Trans. Speech Audio Process.*, 6(3):287–292, May 1998.

[6] J. D. Gibson, B. Koo, and S.D. Gray. Filtering of colored noise for speech enhancement and coding. *IEEE Trans. Signal Process.*, 39(8):1732–1742, August 1991.

[7] A. Yasmin, P. Fieguth, and Li Deng. Speech enhancement using voice source models. In *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, volume 2, pages 797–800, March 1999.

[8] Z. Goh, K.-C. Tan, and B. T. G. Tan. Kalman-filtering speech enhancement method based on a voiced-unvoiced speech model. *IEEE Trans. Speech Audio Process.*, 7(5):510–524, September 1999.

[9] V. Grancharov, J. Samuelsson, and B. Kleijn. On causal algorithms for speech enhancement. *IEEE Trans. Audio, Speech, Lang. Process.*, 14(3):764–773, May 2006.

[10] S. So and K. K. Paliwal. Modulation-domain Kalman filtering for single-channel speech enhancement. *Speech Commun.*, 53(6):818–829, July 2011.

[11] H. Hermansky, E. A. Wan, and C. Avendano. Speech enhancement based on temporal processing. In *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, volume 1, pages 405 –408, May 1995.

[12] J. G. Lyons and K. K. Paliwal. Effect of compressing the dynamic range of the power spectrum in modulation filtering based speech enhancement. In *Proc. Interspeech Conf.*, pages 387–390, September 2008.

[13] Mike Brookes. The matrix reference manual. http://www.ee.imperial.ac.uk/hp/staff/dmb/matrix/intro.html, 1998-2012.

[14] P. Kabal. Ill-conditioning and bandwidth expansion in linear prediction of speech. In *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, volume 1, pages I–824 – I–827, April 2003.

[15] D. M. Brookes. VOICEBOX: A speech processing toolbox for MATLAB. http://www.ee.imperial.ac.uk/hp/staff/dmb/voicebox/voicebox.html, 1998-2012.

[16] J. S. Garofolo. Getting started with the DARPA TIMIT CD-ROM: An acoustic phonetic continuous speech database. Technical report, National Institute of Standards and Technology (NIST), Gaithersburg, Maryland, December 1988.

[17] H. J. M. Steeneken and F. W. M. Geurtsen. Description of the RSG.10 noise data-base. Technical Report IZF 1988–3, TNO Institute for perception, 1988.