Real-time acoustic beamforming on a PC

T.J.MOIR
School of Engineering
Institute of Information and Mathematical Sciences
Massey University at Albany
Auckland
NEW ZEALAND
t.j.moir@massey.ac.nz

Abstract: - A two-microphone adaptive beamformer is implemented in real-time on a personal computer. The language used is National Instruments LabView. This data-flow language is ideally suited for the rapid prototyping of this kind of application and uses two normalized least-mean squares (NLMS) algorithms together with a special voice-activity detector (VAD). The VAD defines an active zone directly in front of the microphones within which valid speech is assumed. Outside of this zone is assumed to be noise of any description. Typical applications of the method is for hearing aids or noise reduction in speech recognition systems.

Key-Words: - Beamformer, Adaptive filter, Speech enhancement, Real-time systems

1 Introduction

The beamforming problem is a topic which has been studied for some thirty years and has application to such areas as communications [1], hearing aids [2], speech-recognition [3] robotics [4] and hands-free telephony [5]. The problem considered here is to use a real-time beamformer to reduce the effects of noise on a speech signal. If the noise can be isolated from the speech then a two microphone approach [6] can be used with one microphone near the desired speech and a second microphone near the noise source. The resulting adaptive filter is updated using the least-mean-squares algorithm (LMS) [7]. This approach is only successful if the speech signal is far enough away from the noise so that elements of the speech are not picked up by the noise microphone. In fact good coherence is required for the algorithm to work and this necessitates the microphones to be close together whilst they also need to be far apart so that the signal is not picked up by the noise microphone and subsequently cancelled along with the noise.There may be certain environments where this approach works but in many realistic real-world situations it is recognised that other more refined methods are required.

A better approach is to keep the two (or more) microphones close together and update the LMS algorithm only during noise and to freeze the LMS algorithm otherwise and keep the last weight vector updated during noise alone. Although this technique overcomes the previous problems encountered above, this improved method now requires a voice-activity detector (VAD). Should the VAD fail to register speech when it occurs then this approach will treat any speech like noise and cancel it too. Therefore the essence of good cancellation when the microphones are close together is that of a robust VAD. The particular type of beamformer used here is a modified version of that of Griffiths and Jim [1]. An improved version of this work has been studied by Van Compernolle [8] where two LMS algorithms are used (for two microphones). The first LMS is updated only during speech and acts as an adaptive beam-steering filter whilst the second LMS is updated only during the noise and acts as the filtering algorithm. Of course the true speech is never isolated from the noise otherwise there would be no filtering problem in the first place but rather the noise power of the speech is assumed to be greater than that of the noise and this activates the steering algorithm. This particular algorithm has been applied to the hearing impaired with some encouraging results [9].

The algorithms discussed here, are implemented on a typical high-performance dual-processor personal computer. The language used is LabView which is the industry standard for instrumentation. The importance of a great deal of graphical output cannot be underemphasised for this sort of development work and as a learning tool. The program enables real-time monitoring of time-domain and frequency-domain parameters together with recording and play-back facilities. The work is

intended as a precursor to implementation on DSP processors.

2 The beamforming algorithm.

Consider the switching algorithm originated by Van Compernolle and Leuven. A block diagram of the particular case of two microphone input is shown in Fig. 1 below.

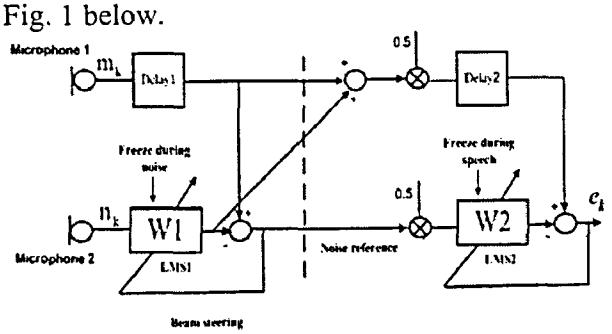

Figure 1. An adaptive beamformer.

The idea of having two rather than one LMS algorithm is to provide a signal-free noise reference for the LMS2 algorithm illustrated in Fig. 1. The error of LMS1 feeds into the noise reference of LMS2. If LMS1 is steered towards the speaker with the noise coming from a different direction than the speech component appearing in the reference should be minimal. The traditional approach has been to either talk directly in front of the two microphones and hope that the delay to each microphone is small and similar (hence the difference will be zero) [10] or to calculate the time-delay to each microphone and compensate for the time-difference of arrival (TDOA).

The trouble with the latter approach is that rarely if at all is the acoustic transfer function of the speech to each microphone a pure time-delay. In a real environment there are reverberations and the acoustic transfer function will be something more complex, a pure delay plus a (possible) non-minimum phase transfer function. This is why the LMS1 steering algorithm is included, to compensate for the transfer function difference of arrival instead of the TDOA. The two time-delays (Delay1 and 2) are to provide physical realisability when there is the possibility of an uncausal solution if the microphones are in the wrong position with respect to each other or with non-minimum phase acoustic transfer functions.

The popular LMS algorithm has trouble with stability for many real-time applications where the signal and noise are non-stationary. For an error signal, primary signal, weight vector and regression vector (composed of past values of reference noise signal), ordinary LMS is given by [7].

$$e_k = s_k - W_k^T X_k \tag{1}$$

$$W_{k+1} = W_k + \mu X_k e_k \tag{2}$$

However, the step size μ will often be either too large or too small. Too small and the convergence is too slow, too large and there is a good chance of instability for large dynamic ranges. This is because for convergence in the mean-square $\mu < 1/\sigma^2$ where σ^2 is the variance of the reference noise signal [11] which more than often is non-stationary with a wide dynamic range. The modified LMS algorithm known as normalised LMS does not suffer from any of these problems in real-time. Normalised LMS is given by (1), and (2) is modified accordingly to be

$$W_{k+1} = W_k + \frac{\bar{\mu} X_k e_k}{\delta + \|X_k\|^2} \tag{3}$$

where δ is a small positive constant that prevents division by zero for small X_k.The algorithm converges in the mean-square provided $0<\bar{\mu}<2$. Good real-time results were obtained for $\bar{\mu} = 0.5$ with no instability problems. In order for the algorithm to steer towards the desired speech, LMS1 must be adapted during periods of active speech whilst LMS2 must be adapted during periods of noise with no speech present. This leads to the inclusion of a voice-activity detector.

3 The voice-activity detector (VAD)

The VAD is crucial to the overall performance of the beamformer. For instance if LMS2 is updated during an instance of speech rather than noise then the speech will be attenuated along with the background noise. The VAD must therefore be capable of switching on rapidly when speech occurs and switching off just as rapidly during the noise periods. Probably one of the simplest ways to do this is to work with thresholds of energy or power and to make a decision by trial and error. With such an approach the VAD needs to know what the ambient background noise level is in the first place so that any speech signal will be flagged if it has a power much greater than the noise. Of course such an approach will only work for positive signal to noise ratios but may well be sufficient for a great many applications. An alternative more robust approach would be to confine the speech to a particular area directly in front of the two microphones and to assume that any noise comes from a different direction based on time-delay estimation and coherence[12]. This latter approach is used here.

The following algorithm is based on the generalised cross correlation method (GCC) and is a robust method of estimating time-delay. The time-difference of arrival (TDOA) is calculated using the GCC and this used to determine whether desired speech is present directly in front of the two microphones. It is assumed that in a great many applications that the desired speech will be in a zone directly in front of the microphones and that the noise will be outside of this zone. Hence if the TDOA is calculated to be greater than a fixed amount (depending on chosen the size of the zone) then the signal is assumed to be noise, otherwise speech. The zone can be shown to be outside of a two-sheet hyperboloid.[12].To avoid problems with reverberation (eg if a noise source outside of the zone reflects back off a wall so that it appears in the zone itself giving a false reading) the magnitude-squared coherence function (MSC) is used. It is known for instance that a reverberant signal has smaller coherence than a direct-path signal. The condition for desired speech in the VAD is therefore that the TDOA be less than some pre-defined value (normally 5 samples) and that the averaged coherence be greater than some fixed amount (normally 0.3).

The setup of the VAD is shown in Fig.2 below. The two cone-like halves of the Hyperboloid represent regions where the delay is a constant. The space in between these 'cones' is the active zone where desired speech is presented. This extends behind the microphones as well as above and below but presents no real problem as in a confined enclosure only the forward section will be active. Such an enclosure could look very much like a telephone kiosk for example and would have a foam type backing to reduce reverberations.

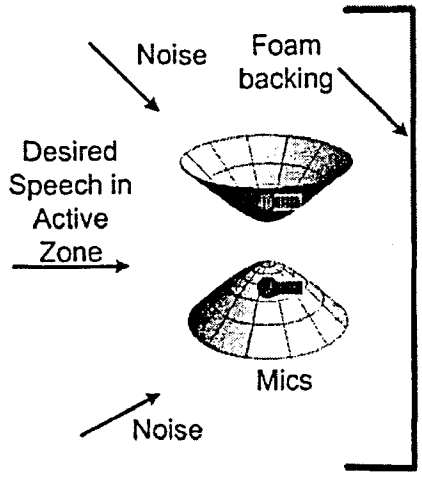

Figure 2. Top: Showing active zone where desired speech is presented.

VAD Algorithm

Step 1: At each FFT frame index i=1,2,3,... assign the two time-domain vectors from each microphone as $x_k = [n_0, n_1, ...n_{N-1}]^T$, $y_k = [m_0, m_1, ...m_{N-1}]^T$

with corresponding frequency vectors obtained from the FFT as $X_j(i)$ and $Y_j(i)$, j=0,1,2...N-1. respectively. It is assumed that the time-domain signals have been suitably windowed before applying the FFT algorithm.

Estimate the spectra (periodogram estimates) of the signals from each of the two microphones:

$$\hat{S}_{nn}(i) = \beta\hat{S}(i-1) + (1-\beta)X(i)X^*(i) \qquad (4)$$

$$\hat{S}_{mm}(i) = \beta\hat{S}(i-1) + (1-\beta)Y(i)Y^*(i) \qquad (5)$$

(4) and (5) is a method of smoothly updating the spectrum recursively at each FFT frame rather than a straight batch method. In the above equation '*' represents complex conjugate and $0 \le \beta < 1$ is a forgetting factor. For the results used in this paper $\beta = 0.5$ was used as a compromise between fast tracking and smoothed periodograms. If β is chosen to be too large then the tracking ability of the GCC time-delay estimator is severely compromised. Some experimentation is required depending on the application.

Step1:Estimate the cross-spectrum (cross-periodogram) from:

$$\hat{S}_{nm}(i) = \beta\hat{S}(i-1) + (1-\beta)X(i)Y^*(i) \qquad (6)$$

Step2 :Estimate the MSC at each FFT frame from:

$$|\hat{\gamma}_{nm}(i)|^2 = \frac{|\hat{S}_{nm}(i)|^2}{\hat{S}_{nn}(i)\hat{S}_{mm}(i)} \qquad (7)$$

and at each frame i, average over frequency k the MSC thus

$$|\bar{\gamma}_{nm}(i)|^2 = \sum_k |\hat{\gamma}_{nm}(i)|^2 \qquad (8)$$

Step3: Estimate the term $\psi_g(i)$ from

$$\psi_g(i) = \frac{|\hat{\gamma}_{nm}(i)|^2}{|\hat{S}_{nm}(i)|[1-|\hat{\gamma}_{nm}(i)|^2]} \qquad (9)$$

Step4: Estimate the time-difference of arrival ℓ (samples) from the inverse FFT of the generalised cross-correlation:

$$\hat{R}^{g}_{nm}(\ell) = \max F^{-1}\{\Psi(i)\hat{S}_{nm}(i)\} \qquad (10)$$

That is, the maximum of the inverse FFT of $\Psi(i)\hat{S}_{nm}(i)$ is the time-delay in samples. A positive delay can be inferred if the maximum occurs in the region $0<\ell<N/2-1$ ie the first half of the inverse FFT and a negative delay if the maximum occurs in the upper half of the inverse FFT.

Valid speech is then assumed when for some zone-limit integer delay ℓ_{max}

$$\text{Estimated delay } \ell \le \ell_{max} \qquad (11)$$

And when the averaged MSC is greater than the MSC threshold

$$\left|\overline{\gamma}_{nm}(i)\right|^2 \ge C_{min} \qquad (12)$$

For the experiments carried out in this paper a sampling interval of 22050Hz was used so that each sample interval corresponds to $T_s = 45.35\,\mu s$. Typically ℓ_{max} was chosen to be around 5 samples and C_{min} was chosen as 0.3.

An illustration of how the GCC appears under normal working conditions is shown in Fig. 3. This is a 'snapshot' since the real GCC is time-varying.

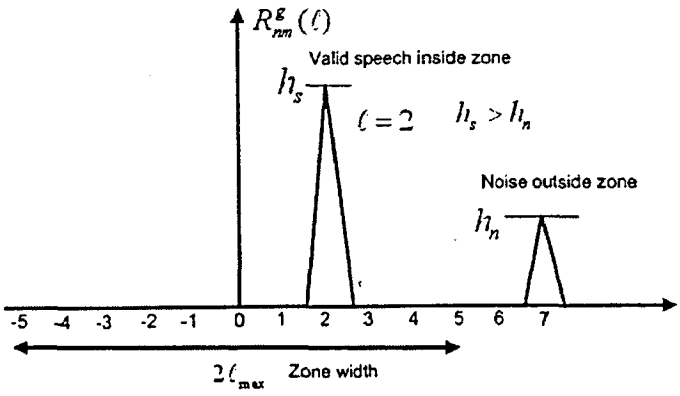

Figure 3 GCC for positive SNR and positive delay.

It is assumed that valid speech has more power than the noise and so the maximum appears for this example at $\ell = 2$. For negative delay any peaks occur in the negative side of the graph. Of course the signal could have positive delay and the noise negative delay depending on the geometry of the problem.

For the special case when the SNR is negative it was found that the noise is so loud that for most practical purposes the case would not exist in say an office environment. However for some factory or military applications it is quite possible that such a condition would exist. Therefore the case for negative SNR is shown in Fig 4.

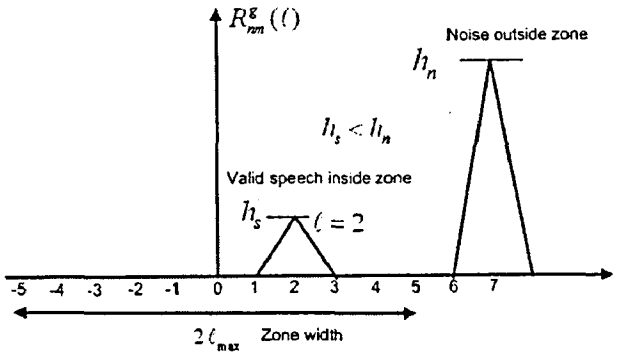

Figure 4 GCC for positive SNR and positive delays.

For this case it can be seen that the maximum occurs outside of the zone since the power of the noise is bigger. A false reading can occur ($\ell = 7$) for this case. To overcome this problem the maximum of the GCC within the zone itself needs to be calculated and compared with the overall maximum. If it turns out to be significantly smaller then it is assumed that no speech is present otherwise speech is assumed at the maximum within the zone and the reading outside the zone ignored. Of course this method will not be always guaranteed to work but is a good compromise for this difficult case.

4 The design of the virtual instrument

The beamformer virtual instrument is written using the programming language 'g' (LabVIEW data-flow). LabVIEW is particularly suited to this sort of application as it was designed specifically for real-time instrumentation applications. The speech signals were sampled using an external USB sound card (for lower noise) though any sound card could have been used. The microphones used were 30cm apart and were both omni-directional magnetic microphones which needed further pre-amplification before feeding to the sound card. The sampling frequency was chosen to be 22050Hz with 16 bits/channel. This gave quite a high quality performance with a Nyquist frequency bandwidth of

around 11kHz. In all of the tests the beamsteering LMS1 used 100 weights with a delay of 5 whereas the main noise-cancelling LMS2 used 600 weights with a delay of 50.(see Fig 1).

The front panel shown in Fig.5 of the beamformer virtual instrument consists of various displays and switches used to evaluate the algorithm. The front panel is too big to show in any level of detail but individual displays will be shown to illustrate the various functions.

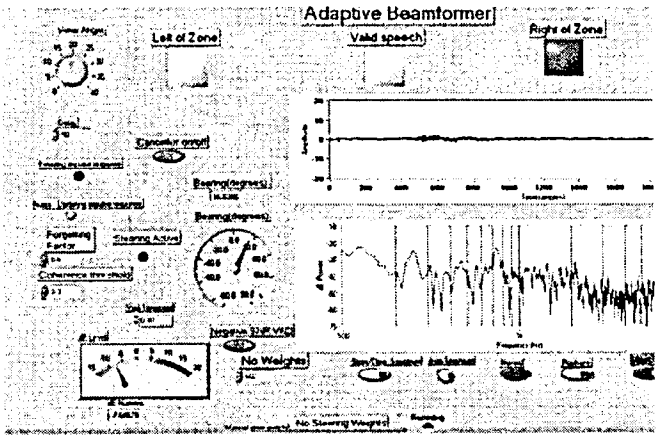

Figure 5. Front panel of virtual beamformer.

For instance a graph of one of the microphone signals versus time is available and below it, the real-time spectrum of the noisy or the enhanced speech. The ability to switch on and off the beamformer was crucial so as to see the dB improvement. The error for LMS2 is the enhanced speech signal and was fed to the sound card so that the results could also be heard in real-time (with a short latency). Indictors (large LED type displays) were used to indicate if a signal is within or to the left or right of the active zone. The size of the zone can be easily altered by changing $\ell_{\max}$.

Bearing Estimation

Although the bearing of a signal was not required for this beamforming algorithm it was decided to calculate and display it for evaluation purposes. This was done by using the TDOA as illustrated in Fig. 6.

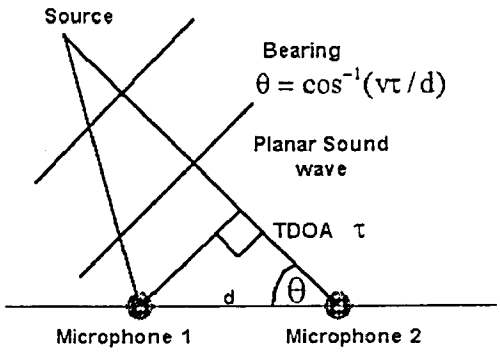

Figure 6. Bearing calculation from TDOA and distance between microphones d=30cm. The velocity of sound is v.

It is of course not possible to distinguish whether the signal source arises from in front of or behind the plane of the microphones. However, we assume a foam backing and that little or no sound arises behind the microphones. The sign of the angle is easily found as it is directly related to the sign of the TDOA as found from the GCC method. So it can be determined from which side of the microphone plane a signal is coming from. A meter centred on zero and calibrated in degrees is used for this purpose. After implementing this method it was found that the angle measurement system was too fast and that it required some dynamics to slow the indicating needle down. This was accomplished by adding a first order time-constant. Of course using the basic equation as in Fig 4 it is also possible for a given $\ell_{\max}$ to calculate the active zone for the VAD. This is found from $\theta_z = \sin^{-1}(v\tau_{\max} / d)$ where $\tau_{\max}$ is the threshold TDOA which defines the zone and d is the distance between the microphones=0.3m. Here $\tau_{\max}$ is calculated for a sampling interval T_s as $\tau_{\max} = \ell_{\max} T_s$. Hence for $\ell_{\max}$=5 the angle is +/- 14.44 degrees with respect to an axis which divides the microphones. (that is directly in front of the microphones represents zero degrees viewing angle). In the program we use the opposite – defining the active zone and calculating $\ell_{\max} = \frac{d}{vT_s}\sin(\theta_z)$ which is fed directly into equation (11).

Measurement of dB power

Assuming a signal which is being measured has zero dc (which is the normal case with audio signals) then its variance (or average power) can be calculated recursively from results in [13] and a dB meter (shown in Fig 5) can be used to measure

overall dB noise reduction. Using the results of [13] we have a dB meter block diagram as illustrated in Fig. 7.

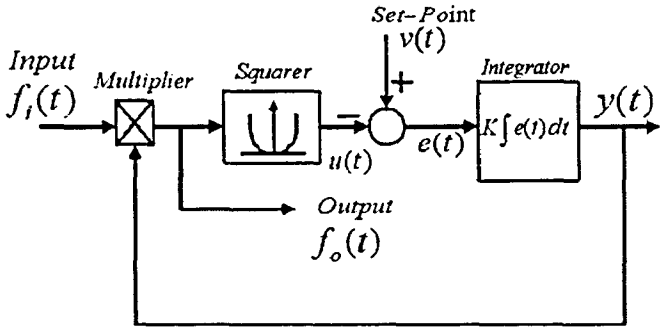

Figure 7. Automatic Variance Control

This method is covered in some detail in the reference and has been called 'Automatic Variance Control'. For any dc-free input signal $f_i(t)$, then by virtue of negative feedback and high loop-gain, its reciprocal power (variance) 1/p(t) is tracked in real-time as shown by y(t) in Fig.7. The diagram is continuous time but the discrete-time version is easily found by replacing the integrator with its z-transform equivalent. The set-point is fixed at unity whilst the output $f_o(t)$ is not required for this application. Since the desired output is power and not reciprocal power, then y(t) must be inverted. However, since when using dB we have $10Log_{10}(1/p(t)) = -10Log_{10}p(t)$ then the power is obtained by converting y(t) to dB and changing the sign. A theoretical limitation occurs when the noise-power is zero but in a real environment ambient noise of some finite power is always present.

The bearing angle indicator and the noise-power dB meter is illustrated in Fig 8..

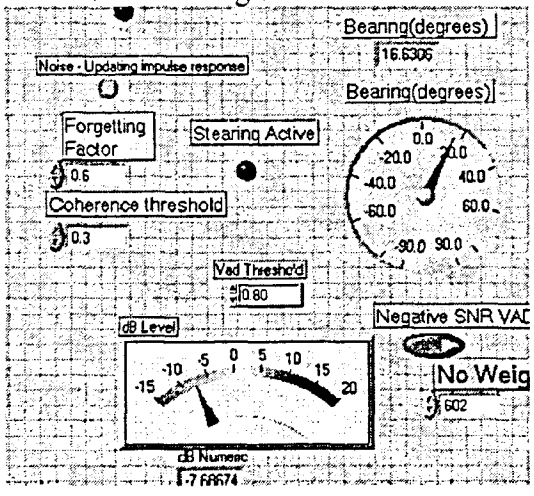

Figure 8. Bearing indictor and dB level meter.

It is also possible to use Fig 7 as a method of automatic gain-control (AGC) for each of the two microphone inputs. For AGC the output $f_o(t)$ is used. However, AGC has the problem that it can amplify the noise when there is little signal or noise. It is successful in preventing saturation when the power of the signal+noise is outside the dynamic range. Therefore the AGC is left as an option.

5 Experimental results

Experiments were carried out in a typical office environment 4m by 4m. In all experiments a word had to be spoken first so as to steer the beam in the 'look' direction which in this instance was directly in front of the microphones. When the beamformer was switched on, the effect was quite dramatic and a comparison of the average spectrum before and after beamforming showed a reduction of the base-level noise (with no speech signal present, only ambient fan noise from the PC) right across the spectrum up to the Nyquist frequency. This is illustrated in Fig 9.

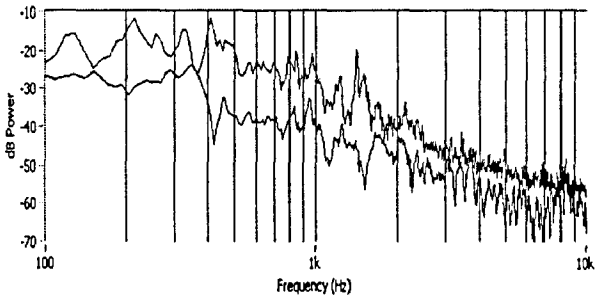

Figure 9. Top: Original spectrum of ambient noise,bottom:spectrum with beamformer.

Frequencies of less than 2.5kHz give the best performance with reductions of up to 10dB or more at some frequencies and as much as 30dB reduction at 400Hz. The area under the spectra gives rise to the total average power of the noise and can be measured in real-time using the method shown in Fig 7.

A radio 1m away was used to provide background noise. Fig.10 illustrates typical results that were obtained. The words 'one','two',three' were spoken and repeated with the beamformer turned on. In Fig. 5 the beamformer was switched on at around sample no 100,0000 (mid way on the graph). The VAD flag can be seen around the desired words and the background speech is attenuated by around 10dB without any noticeable reduction in quality of the desired speech.

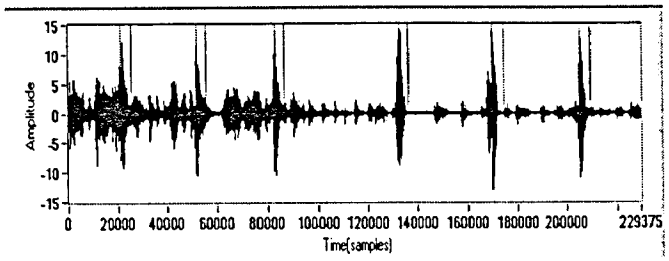

Figure 10. Performance without and with adaptive beamformer. (speech + speech).VAD flag also shown.

To show the tracking ability of the beamformer a radio was presented some half a metre directly in the active zone of the VAD and then moved quickly to the left of the zone at a similar distance. The radio noise was attenuated as shown midway through the time axis of Fig 11.

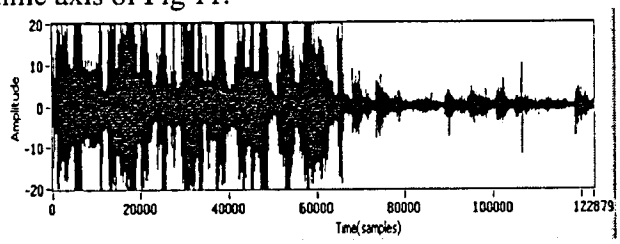

Figure 11. Performance with a radio in and out of the VAD active zone.

This shows a dramatic attenuation when measured of approximately 9.6dB. Similarly the spectrum is shown for the same experiment in Fig.12. The top graph is the average spectral density within the VAD active zone and the bottom outside of this zone.

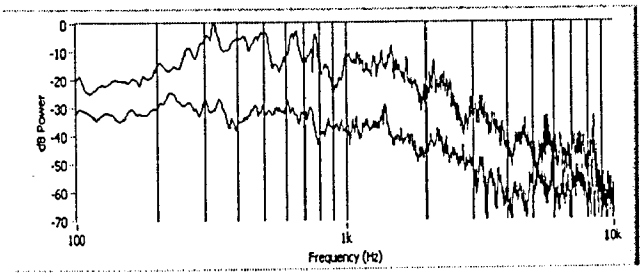

Figure 12. Spectral performance with a radio moving in and out of the VAD zone.

The reduction in noise power is apparent right across the spectrum and at a good many frequencies it is as high as 20dB attenuation. However, the dB meter indicated an average overall dB reduction in noise of around 12dB for most applications.

Finally, a polar plot was obtained in the same environment using white-noise as the source and by measuring the dB reduction using the built-in dB meter. The angle was measured using the previously discussed bearing estimator and a distance of one metre was used for the source. Fig 13 shows one half of the plot (the plot from the rear being omitted). It can be seen that the active zone shows up well and it's size can easily be altered.

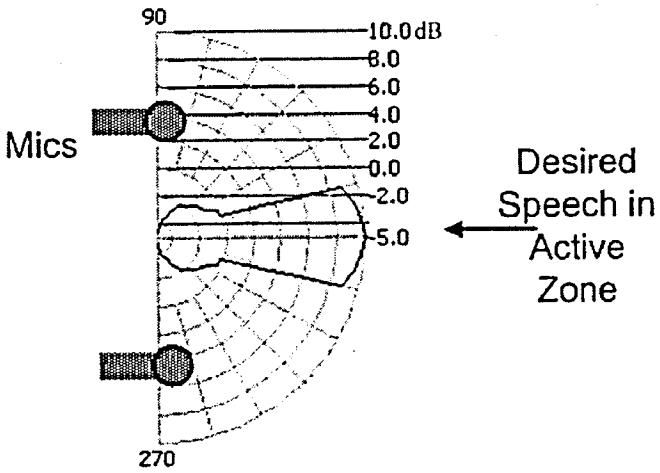

Figure 13 Polar plot of beamformer

6 Conclusion

An acoustic two-microphone beamformer has been implemented in real-time as a virtual instrument. In this way the algorithm, which is an extended switching Griffiths-Jim beamformer can be closely examined. The performance was studied with speech and speech plus interfering noise. The combination of the robust VAD using GCC and the dual NLMS approach gives rise to a powerful method for real-time noise-reduction evaluation.

References:

[1] L. J. Griffiths and C. W. Jim, "An alternative to linearly constrained adaptive beamforming," *IEEE Transactions on Antennas and Propogation*, vol. 30, pp. 27-34, 1982.

[2] B. Widrow and F. Luo, "Microphone arrays for hearing aids:An overview," *Speech Communication*, vol. 39, pp. 139-146, 2003.

[3] T. Nishiura, R. Gruhn, and S. Nakamura, "Collaborative steering of microphone array and video camera toward multi-lingual tele-conference through speech-to-speech translation," *presented at Automatic Speech Recognition and Understanding*, 2001. ASRU '01. IEEE Workshop on, 2001.

[4] S. Stergiopoulos and A. C. Dhanantwari, "Implementation of adaptive processing in

integrated active-passive sonars with multi-dimensional arrays," *presented at Advances in Digital Filtering and Signal Processing*, 1998 IEEE Symposium on, 1998.

[5] G. W. Elko, "Microphone array systems for hands-free telecommunication," *Speech Communication*, vol. 20, pp. 229-240, 1996.

[6] B. Widrow, J. R. Glover JR, J. M. Mc Cool, J. Kaunitz, C. S. Williams, R. H. Hearn, J. R. Zeidler, E. Dong JR, and R. C. Goodlin, "Adaptive noise cancelling:Principles and applications.," *IEEE Proceedings*, vol. 63, pp. 1692-1716, 1975.

[7] B. Widrow and M. E. Hoff, "Adaptive switching circuits," *IRE Wescon Convention Record*, pp. 96-104, 1960.

[8] D. Van Compernolle and K. U. Leuven, "Switching adaptive filters for enhancing noisy and reverberant speech from microphone array recordings," *presented at Proceedings of the IEEE International conference on acoustics,speech and signal processing*, Albuquerque, 1990.

[9] J. Vanden Berghe and J. Wouters, "An adaptive noise canceller for hearing aids using two nearby microphones," *Journal of the Acoustical Society of America*, vol. 103, pp. 3621-3626, 1998.

[10] D. R. Campbell and P. W. Shields, "Speech enhancement using sub-band adaptive Griffiths-Jim signal processing," *Speech Communication*, vol. 39, pp. 97-110, 2003.

[11] S. Haykin, *Adaptive filter theory*. Englewood Cliffs New Jersey: Prentice Hall, 1986.

[12] H. Agaiby and T. J. Moir, "Knowing the wheat from the weeds in noisy speech," *presented at Eurospeech*, Sept 1997,Rhodes,Greece

[13] T. J. Moir, "Automatic variance control and variance estimation loops," *Circuits Systems and Signal Processing*, vol. 20, pp. 1-10, 2001.