Acoustic Echo Cancellation in a Reverberatory Chamber Using an Adaptive Least Means Square Algorithm

Vincent Allen, Mark E Henderson, Erik S Wheeler, and John Williams

The Echo Cancellation Group
Department of Electrical and Computer Engineering
Mississippi State University
Mississippi State, Mississippi 39762
{allen, henders, wheeler}@isip.msstate.edu and {SPwillia@ee.msstate.edu}

ABSTRACT

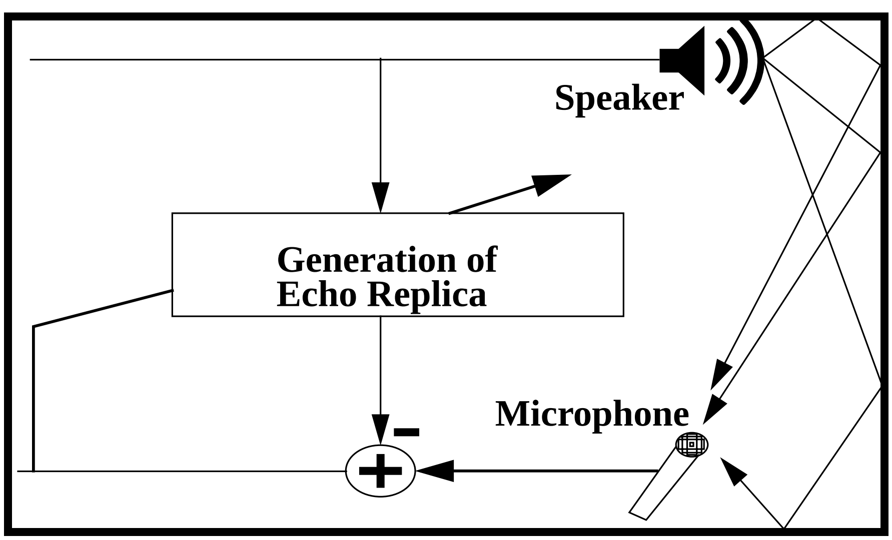
In an increasingly noisy society, methods of reducing noise are becoming more important. Noise, or unwanted sound, may be reduced in an environment by two basic means: passive noise control and active noise control (ANC). Active noise control is a method of reducing noise by canceling a sound wave with an inverted copy of itself. This process works best in a simple environment: one in which the wavelength of the noise is long in relation to the dimensions of the space. ANC has been most successful in reducing noise in ducts and headphones (essentially one dimensional problems). This project centered around applying ANC techniques to reducing reverberatory echoes in collected data. The models built during this study are, in effect, one dimensional spaces were ANC is applied.

1. INTRODUCTION

In the Electrical Engineering building at Mississippi State University, Simrall, there are three large, four story stairwells. These stairwells are extremely reverberatory, and sustain a low frequency noise for long periods of time (~10 seconds). The reverberation is, in fact, so bothersome that conversations must be suspended while traversing the halls because the conversation quickly deteriorates. The frequencies amplified by the room are mostly at the lower end of the audio spectrum, as would be expected from a large cavernous room.

There are several ways that the reverberating action of a large space can be reduced. One option is to install baffles and other types of physical damping to the room. Using physical methods is most appropriate for high frequencies since the size and weight requirements of low frequency baffles tend to outweigh their usefulness. For example, an ideal location for global ANC is airplane cabins where the noise is very bothersome and in some instances medically harmful. In this case, the weight of the necessary baffles would severely restrict the capabilities of the airplane.

This project focused on echo cancellation, starting with the elimination of a computer-simulated reverberation. This first stage consisted of the development of versatile echo cancellation code using the least mean square adaptive filtering algorithm described below. Adaptive filtering examines an input, attempts to cancel it, and adjusts the filter coefficients of an FIR filter to compensate for the error. This code was tested by inputting artificial sound files with simple impulse responses, and canceling the echo through adaptive filtering techniques.Upon completion of the simulated echo cancellation phase, the parameters were modified to handle real signals recorded in the Simrall stairwell. These modifications consisted mainly of increasing the tap length and reducing the error correction step size.



The canceller was not tested in a real time experiments this semester due to time constraints. But if it were, the reduction in noise would only be evident within a foot or two of the speaker, depending on the wavelength of the signal. This is due to the single input-single output, one-dimensional filter we are designing. Since the signal to be canceled will not follow the same path as the canceling signal, the 160 - 200 degree phase difference required for cancellation will only occur at particular locations in the room. It should be noted that, due to the varying phase difference, the signal will also be amplified in some areas. This problem could be controlled by the use of additional sensors and speakers.

A major difference between active acoustic echo cancellation and the echo canceller currently in wide-spread use is that the echo canceller has the reference signal at hand, thus making an echo estimate a relatively trivial task, namely one of simple convolution, which can then be used to filter out the unwanted echoes. To make the link between these two it is necessary to develop the inverse filter corresponding to the FIR filter developed by the algorithm. Such a filter could be used to continuously filter incoming data without knowing the reference signal, but this step in the procedure is not possible given the problem constraints which are strictly enforced by Father Time.

2. TECHNOLOGY TODAY

2.1. Hardware/Software:

Since there are many different applications for echo cancellation, there are naturally many different types of hardware and software available to the DSP engineer. Most of the echo cancellation software and hardware developed thus far has been developed primarily for telecommunications. Echo cancellation on the phone lines has been necessary for many years, due to the relatively long signal paths introduced by satellite and long terrestrial connections, to provide quality service. Other uses for echo cancellation technology include new and rising fields such as audio/video conferencing.

Texas Instruments(TI) is a major provider of digital signal processors, namely the TMS320 family. This family includes both dedicated and programmable 16-bit fixed-point and 32-point floating point DSPs. Application specific DSPs available from TI include audio/video applications that implement industry standards (MPEG, Dolby, etc.). Other possible uses include active noise cancellation, motor control, computer components, and consumer electronics.

Most echo cancellation is done using a programmable DSP such as the TMS320C3x. Several companies have developed software to implement various echo cancellation algorithms. Philips Kommunikations Industrie AG has TMS320C3X Digital Signal Processing software which implements the CCITT standard for Acoustic Echo Control and allows full-duplex communication for audio/video conferencing. DSP Software Engineering, Inc. provides software for the implementation of an Audio Line Echo Canceller on the TMS320C3x. AT&T also provides the QuietQuiet™ Acoustic Echo Cancellation (AEC) software as an implementation of there very own patented echo cancellation technology.[20]

2.2. Benefits/Performance[20]

QuietQuiet™

* Environments range from small cubicles to large conference rooms.

* Provides high-quality, full-duplex

* Adaptively cancels acoustic echoes arising in hands-free audio/video teleconferencing systems.

* Environments range from small cubicles to large conference rooms.

* Provides high-quality, full-duplex speech communications typical of dedicated video conferencing systems.

* No switching, dropouts, or speech clipping.

* All parties may be heard simultaneously (double talk).

* Howling rejection.

* Fast, completely automatic training.

* No distracting or extraneous training signals.

* Continuously adapts to changes in room acoustics.

* Continuously adapts to changes in microphone and loudspeaker placement, loudspeaker volume setting, and movement of people.

* Supports both 3.5-KHz and 7-KHz speech communications (G.722 and G.728).

* Subband signal processing architecture minimizes processing load while maintaining high acoustic echo cancellation performance (fast convergence).

* Designed to operate at room gains up to 10 dB, allowing an order of magnitude greater acoustic

power output.

* Audio Processing Bandwidth:

  125 - 3,500 Hz 2 dB, @ 8-KHz sample rate

  125 - 7,125 Hz 2 dB, @ 16-KHz sample rate

* Acoustic Echo Compensation length is determined by host resource availability; varies with frequency
* Convergence Rate of Adaptation: 30 dB/sec
* Adaptive (only) Echo Cancellation: > 45 dB
* Total Echo Cancellation: 60 dB, maximum
* Room acoustic gain: up to 10 dB, nominal
* Software state machine automatically determines each of four states (receive, transmit, double-talk, and idle)

PKI Acoustic Echo Control for the TMS320C3x

* CCITT G.167 compliant
* Bandwidth 300... 3400 Hz / 50... 7000 Hz
* Echo attenuation 45 dB (canceller + center-clipper)
* Full-duplex capability (double talk)
* Cancellation window 256 ms
* SEND path delay 100 ms (option <2 ms)
* Frequency shift 5 Hz
* Line echo canceller for analog lines

2.3. Current Research

In addition to these hardware and software implementations there is still applied research in making the algorithms smarter, less memory intensive, and less computationally expensive.

There are several ways to reduce the computational complexity of echo cancellation algorithms including block adaptive filters, subband filtering, and frequency domain adaptive filters. Both the subband method and the frequency domain method not only are more computationally efficient but also converge faster than the standard LMS algorithm. The problem with each of these methods is that they both introduce a delay in the processing and that they typically lack the ability to track a changing impulse response. In order to overcome these problems, the fast Newton transversal filter algorithm has been recently proposed. The key to better computational efficiency in this approach is the fact that the prediction part of the filter can be of lower order than the size of the filter.

The LMS algorithm has been used extensively in acoustic echo cancellation even though it does not converge very well with speech signals. In cases such as mobile radio where fast convergence is necessary, the FNTF algorithm is better suited. The reason that fast convergence is needed with mobile radio is that the echo path is constantly changing, thus requiring the filter coefficients to be updated continually. However due to the high noise usually encountered in mobile radio environments, the adaptation can only be allowed when there is a high SNR--generally occurring in short bursts. So, a fast converging algorithm will enhance the performance of the echo cancellation.[2]

1. The LMS Algorithm

The LMS adaptive algorithm is well documented in various literature [1-4], hence only a basic introduction to the algorithm is given here--including all changes and important reference notes.

The Algorithm is a simple gradient search algorithm that reduces some cost function, in this case the error between the original signal and the received signal (with the echo). The algorithm is designed to search for the best FIR filter coefficients that represent the echoes in the test space (room, stairwell, etc...). An example of the filter coefficients that are determined by the algorithm and resulting frequency response of the filter are shown in Figure 1.1.

To determine the coefficients, the original reference signal is convolved with the FIR filter, and the result is an estimate of the echoes present in the recorded signal. The difference between the recorded signal and the original signal plus the estimate of the echo is used to determine how the coefficients will be updated

The convolution indicated is not as computationally expensive as it appears. Since time is performing the shift operations, only one set of multiply/adds is needed to determine the echo estimate at the current time. Another time saving shortcut is to limit the number of coefficients updated. For example, if the echo must lag the signal by τ seconds, then a straight delay can be implemented without computing the coefficients over the time period where no echoes can exist.

To further decrease the computational complexity and increase the convergence speed of the algorithm, subsampling could be used. Subsampling is simply splitting the input signal and the output signal into adjacent frequency subbands using analysis filter banks. The impulse response is the system impulse response filtered by the appropriate subband filter. When this approach is used with the LMS algorithm, the convergence speed is increased because the adaptation step size can be set in each subband filter so that it is matched to the energy of the input signal in that particular frequency band.

2. EXPERIMENTAL DESIGN

In order to evaluate the algorithm, several different sets of test data were necessary. Test signals included simple deterministic signals with predefined echoes, complex speech signals with predefined echoes, and simple deterministic signals with complex cavity induced echoes, and complex speech signals with complex cavity induced echoes.

The algorithm was tested extensively on many simple signals, with predetermined echoes, to determine the effects of different parameters on convergence, final SNR, and other relative figures of merit common to such devices. The output of these tests were recorded and analyzed using basic reference tools such as SNR, absolute error, and auditory difference. Upon completion of the preliminary tests the algorithm was tested on real data collected from the stairwell and from the small room adjacent to the stairwell (both rooms yield long-lived echoes).

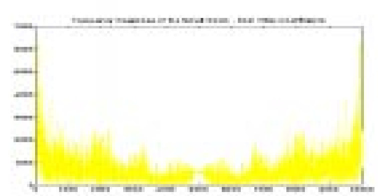
2.1. Experimental Data

The signals with predefined echoes, referred as simulated echo, were produced by convolving original data files with a known filter response. This process produced a perfect echo file in which all components of the signal, including the noise and very low level data, were exactly echoed.

The cavity induced echoes were generated by one of two sources: a four story reverberant stairwell or a small highly reverberant room. To record the echoes produced by these reverberant cavities the following test was constructed. A cart containing a conventional tape player, a DAT recorder, and a microphone was placed as near to the center of the space as feasible. The tape player played prerecorded signals and sounds and the resulting signals were recorded on the DAT at 48 kHz.

Test signals used were a chirp spanning 0 - 20 kHz, a chirp spanning 0 - 500 Hz, a 1000 Hz sine wave, a sum of 100, 1000, and 10000 Hz sine waves, a 100 Hz impulse train, and a single impulse. Upon completion of the test signals a variety of sounds were created in the space and were recorded including speech, footsteps, and door closure (which approximates an impulse). All prerecorded sounds lasted for 30 seconds. After analysis of this data an approximate impulse response of the room was developed. (See Figure 1.1)

Figure 1.1 - Frequency Response of Small Room



2.2. Convergence Tests and SNR

The first and probably most important test of the algorithm is whether or not it converges. Obviously, if the filter coefficients don't converge, the code is useless, so several convergence tests were performed on simulated data. These tests consisted mainly of varying the error correction stepsize, β, and examining the effects on the error, filter coefficients, and SNR plots (Figure 2.1). By viewing these plots, it was often easier to tell whether the filter was properly adapting than it was to listen to the output with the echo removed. To account for variable loop gains the program allows run-time setting of the step size which has a default setting of 2^-10.

2.3. Auditory Assessment

The next test of effectiveness was to listen to the output for the appropriate cancellation of the echoes. This was probably the most satisfying test: hearing the clean output of the echo canceller. As stated earlier, the change was almost unnoticeable on the chirp file played in the room, so this examination was mainly useful for the simulated echoes generated in the speech files,

2.4. Comparison of Final Coefficients

Even though the error might converge and the filter coefficients stabilize, there was no guarantee that these coefficients were a good approximation of the room response. In order to determine this, the code must be run on various signals and the impulse response determined for each case must be examined for conformity. An example of how the coefficients converged in time for a simple echo response is shown below in Figure (***).

A variety of known echoes were used to determine how well the coefficients converged to the actual echo response. A multiple echo response was used as an input, which the algorithm handled perfectly. Then a multiple echo response that extended past the number of taps was used as an input. The filter coefficients slowly converged to the echo response that was contained in the tap length, but the coefficients wobbled much more and took longer to converge.

2.5. Echo Reduction of Unknown Signal

Upon determination of the impulse response, this filter should be able to remove echo from any signal, even without a known reference, containing the same echo characteristics. This was attempted by three different means, the validity of which may be somewhat suspect.

The first attempt was to use the determined frequency response and the FFT of the echo signal to produce the input signal as follows:

$$x(n) = F^{-1}\left\{\frac{Y(f)}{H(f)}\right\} \tag{1}$$

The application of this conversion to the frequency domain and back was too difficult to implement given limited resources (time).

An infinite impulse response (IIR) filter whose coefficients were determined from the output of the echo canceller was created. This filter was implemented using the direct form structure, but unfortunately, this filter was inherently unstable. Stability could have been achieved by reflecting the poles of the filter that were outside the unit circle back into the unit circle (as defined by the z-transform), but such an attempt was not made.

MatLab, which is (in some circles) known for its signal processing toolbox, was employed to deconvolve the filter coefficients out of a signal that had been previously convolved with those (or an approximation to those) filter coefficients. Due to the extreme size and the numerical instability of the deconvolution routine this attempt to remove echoes from an unknown reference also failed.

2.6. Noise

An ever present problem in real-world data collection is 60 Hz transformer, ballast, and power supply hum which significantly altered the results during the speech processing stage of the experiment. Other noise sources encountered in the course of experimentation included wind noise and equipment placement (which severely affects the system transfer function). In the process of playing and recording the data in this environment, noise and other errors quickly accumulate in the data. Hence it is necessary to establish a lower bound on the level of signal that it is feasible to cancel. To perform this function a cutoff parameter is specified, below which, the signal will not be processed. This cutoff also eliminates the case in which the signal is too low to produce an echo.

2.7. System Delay

Inherent in this system setup is some finite delay introduced between the instant the sound is produced and the instant the microphone receives the sound. If the reference file and the echo are not perfectly aligned the filter will try to compensate for the difference. The term in the filter that is introduced by mis-alignment is not desired and can be compensated for by delaying the update function for a predetermined number of samples that is less than the number of samples in which the first echo will occur.

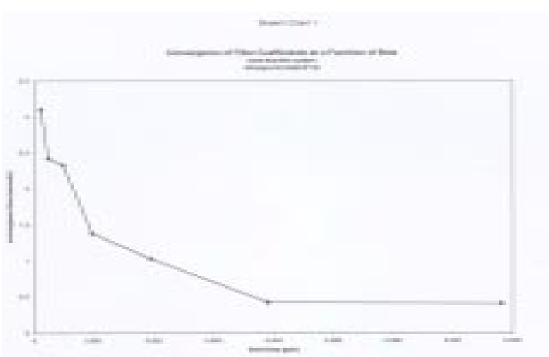
There is another important use of delay. If it is known that a system has an echo length that requires only 200 taps but this echo occurs, perhaps, 1000 taps after the signal, by inserting a straight delay it is not necessary to calculate all of the coefficients of the filter--only those that are necessary to cover the echo length.

3. EVALUATION

3.1. Determination of Step Size

Once convergence of the filter on simulated echo data was confirmed by visual inspection of the error plots, tests were performed on the simulated data to determine appropriate step size, β. Obviously, if the step size is too small the canceller will take a long time to converge on the signal while a large step size may result in a divergent error. Figure 2.1, which shows convergence as a function of β, indicates that as the step size is increased, the convergence time decreases in an inverse relationship. At step-sizes higher than $\beta = 0.008$, the canceller was unstable for the simulated data given. This step size was much smaller for the real data, which would diverge for $\beta > 0.0009$, probably due to the number and the complexity of the echoes.

Figure 3.1 Convergence as a Function of β



3.2. Analysis of Simulated Echo

Before testing the code with real speech data, the code was tested using a simulated echo. This provided a method which allowed the results to be easily compared with the original signal. The output of the echo canceller was compared with the input signal as shown in the following series of Figures. In addition to visual and mathematical comparison (SNR), the input (with the echo), output (echo free), and the original signal were presented to a panel of judges that determined a subjective view as to the extent of improvement provided by the echo canceller. The first signal used to evaluate the echo canceller was a pure speech signal convolved with a triangular filter twenty taps wide with a maximum height of 50% of the original signal and a delay of 2000 taps.

The original speech signal is shown in Figure 3.2, followed by the speech plus echo, Figure 3.3, and the output with the cancelled echo, Figure 3.4. It is clear from this that the canceller trains very well on the echo and almost completely eliminates it after approximately 500 ms. This result can also be viewed on the absolute error plot shown in Figure 3.5 and the signal to noise ratio (SNR) plot shown in Figure 3.6.

Figure 3.2 Original Speech Signal

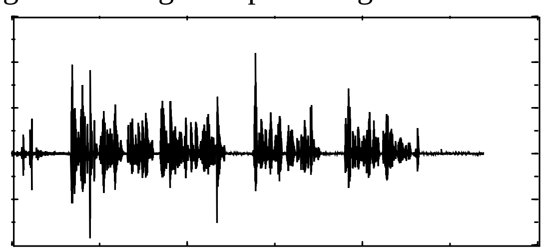


Figure 3.3 Speech Signal plus Echo

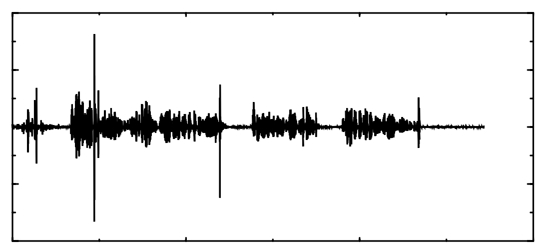


Figure 3.4 Echo Canceller Output (Speech)

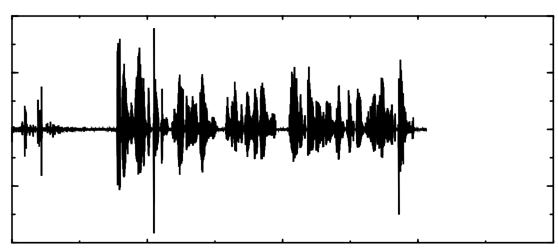


Figure 3.5 Absolute Error (Signal - Output)

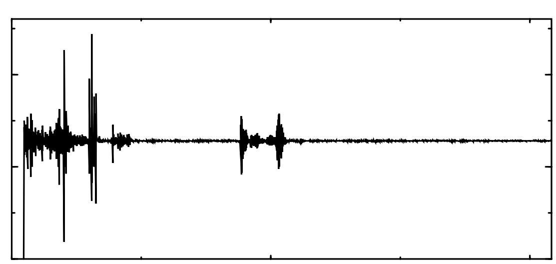
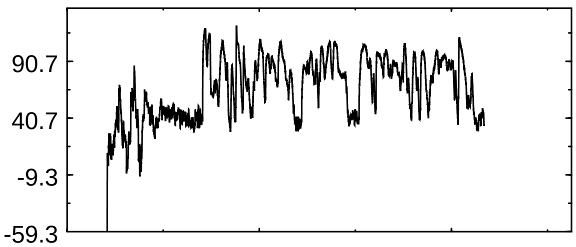


Figure 3.6 Signal to Noise Ratio



Another important consideration in the evaluation of the echo canceller is how the coefficients are being updated. Convergence is directly caused by the values of the filter coefficients and how similar they are to the true filter values. The plots below show how the filter coefficients change in time (Figures 3.7 - 3.9)

Figure 3.7

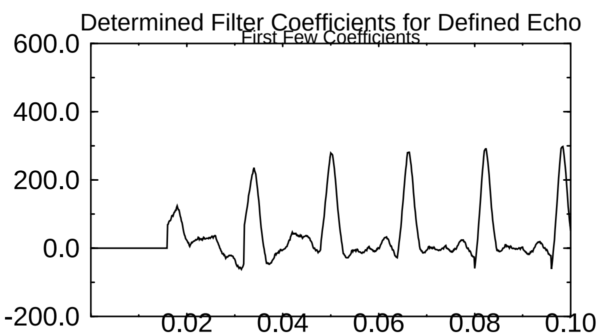


Figure 3.8

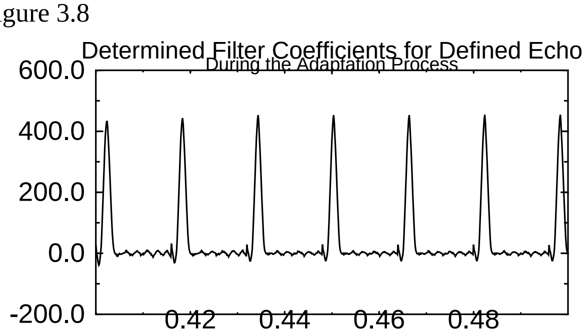
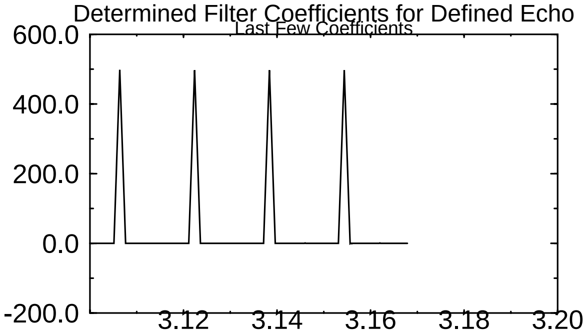


Figure 3.9



The frequency response of the test filter and the filter derived by the algorithm are shown in figures 3.10 and 3.11 respectively. Notice that these two plots are not exactly alike, this difference is due to the zero stuffing that occurred during the frequency transformation of the derived filter.

Figure 3.10 Frequency Response of Original Filter

Figure 3.11 Frequency Response of Derived Filter

3.3. Analysis of Small Room

The echo canceller performs very well on predefined echoes with signal to noise ratios greater than 40 dB. Given the success of the echo canceller for simple signals/simple echoes and complex signals/simple echoes the next stage of the project could be investigated, namely applying these techniques to determine the characteristics of a small room.

The first attempt was made using the low frequency chirp recorded in the small room. In this case, the ecsho recording was done in very close proximity to the speaker emanating the chirp. Therefore, the pure chirp signal dominated the recording, with comparatively low levels of echo. This may have been the reason for the quick convergence of the echo canceller. The plot of error versus time for this chirp signal, shown in Figure 3.7 shows that the canceller takes approximately 11 seconds to converge on the signal. The longer convergence time is due to the small step size required to avoid divergence from this complex echo. The resulting filter approximation of the room at the location of the recording is shown in Figure 3.8.

Figure 3.9 below represents the filter coefficients as a function of time.

Figure 3.7 Chirp Signal Convergence

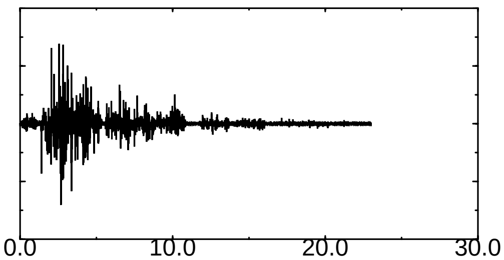


Figure 3.8 Filter Approximation of Room

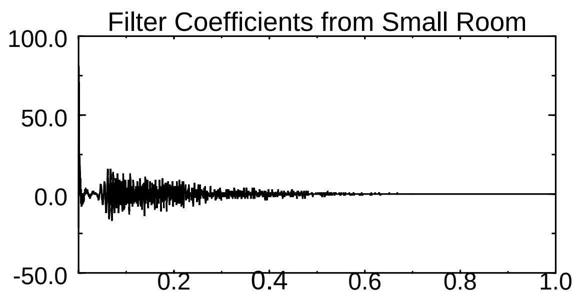
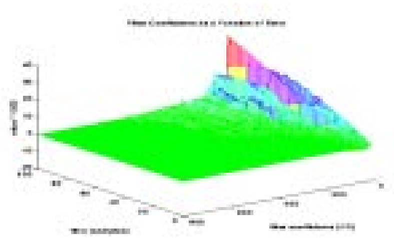


Figure 3.9 Filter Coefficients vs. Time



Convolution of this filter with an undistorted speech signal, gives an approximation of the speech echoed in the room, resulting in a muffled, low volume echo. The chirp was stored with a constant maximum volume of 10000 while the chirp recording has a variety of levels the average of which is about 5000 while the peak is much higher at the resonant frequency of the room. The result of the convolution of this room approximation and the speech file of Figure 3.2 is shown in Figure 3.9. Notice the low level of the signal.

This resultant filter may be a reasonable approximation of the room, but this would not be determined without more testing. In order to confirm the results, one of two tests must be successfully performed: Either the coefficients must be used to deconvolve an echo plus signal into just the signal without knowledge of the reference signal, this will be discussed later, or the canceller must be applied to different data from the room yielding final filter coefficients that are the same as those determined from the chirp.

The method for data collection of the speech signals in the room was, unfortunately, completely different from the chirp. The speech data was recorded with the receiving microphone on the other side of the room from the transmitting device. This echo signal was louder than the original speech file due to a difference in the necessary amplification. We attempted to compensate for this by scaling the echo file such that the undistorted speech in each was approximately the same level. The speech and echo files were also aligned in an attempt to eliminate the delay between the source and receiver. The original speech and the speech plus echo files are shown in Figures 3.10 and 3.11, respectively.

Figure 3.10 Original Speech

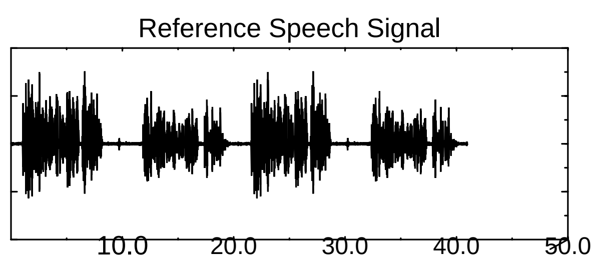
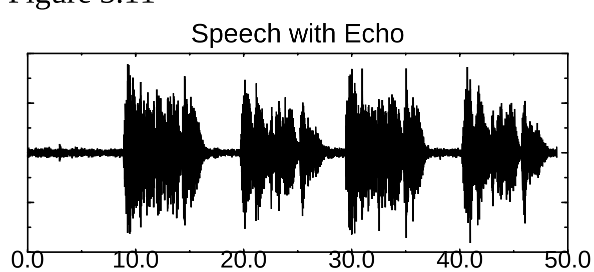


Figure 3.11



These plots clearly demonstrate that the reverberation in this room is extremely high. This complex signal created quite a problem with divergence of the coefficients, requiring a very small step size, B=0.00004, to remain stable.

As can be seen in Figure 3.12, the error never converges, even after 20 seconds. Although mostly useless because of the non-converging nature of the error plot, the resultant coefficients are shown in Figure 3.13.

Figure 3.12

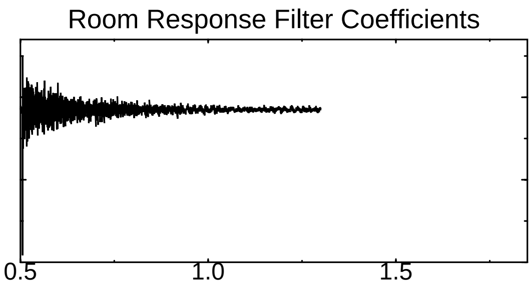
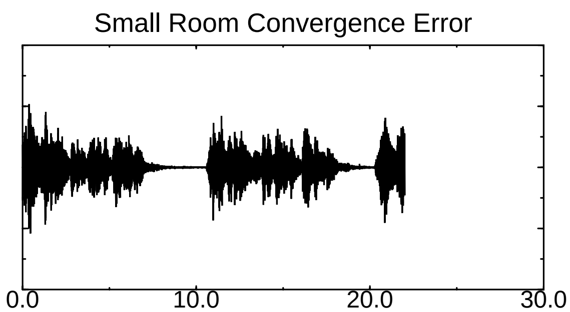
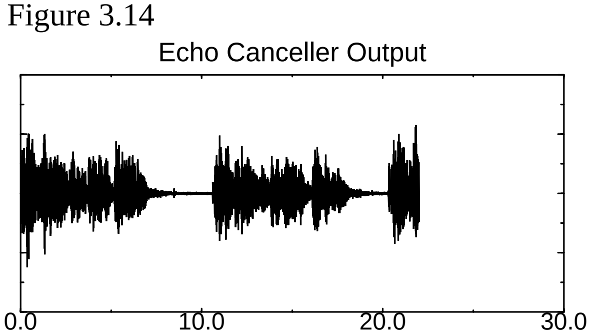


Figure 3.13



Notice that the coefficients seem to be symmetric about the time axis. The echo canceller output, Figure 3.14 for the reference signal shown in Figure 3.10 is probably not a good approximation of this room response since the filter coefficients are dissimilar to those determined from the chirp, Figure 3.8.

Figure 3.14



The results obtained using the chirp reference signal and the speech reference signal are very dissimilar. This difference is due mostly to the difference in how the two trials were conducted. The filter treats the space as a one -dimensional echo cancellation problem, therefore any time the physical setup is changed there is a new one-dimensional problem to solve. In order to determine if the coefficients are accurate, the techniques of signal reconstruction described in Section 3.2 were implemented.

4. SUMMARY

.Most of the project comes down to the implementation of a rather simple, but unexpectedly tricky algorithm. Although, initially, an attempt was made to implement the code using arrays because of their low computational complexity, a successful program using a linked list data structure was finally coded.

This structure allowed for versatility, but increased the run time considerably. Performing the updates using M=1 removed the need for a full convolution for each sample and allowed for only storing the current echo and error. These improvements increased the speed of the code considerably, running at approximately real time for small filters in the range of 128 taps.

The reverberation of the stairwell was too long for the tap lengths required to approximate the response of the room, a considerably longer filter was required, which causes the program to run at a somewhat slower rate. But, if this code were to be implemented in an environment similar to stairwell, hopefully, the response of the room would not change very quickly, and a long training process would be satisfactory. Unfortunately, even after a training period of 30 seconds, the canceller was not able to accurately develop a good filter representation of the small room for speech signals. There are many factors that may have caused this: If the signal file and echo file are not exactly synchronized, the expected signal at a given time will not be present in either the signal or the echo and the canceller will attempt to compensate for this by increasing one of the early coefficients instead of the true echo response. Although the error plot for the chirp seemed to indicate a good room approximation, this may have been due to the low level of the echo in the recording or a synchronization problem, and, as the canceller trained on the first few coefficients, the error decreased without actually approximating the room response.

The fact that the speech data was taken in a different manner than the chirp may also be a contributing factor to the canceller's inability to train, since the canceller assumes that the signal is the same level in the echo file as it is in the original signal file. If this is not the case, the first coefficient of the filter will become the difference of the signal and the echo.

5. ACKNOWLEDGEMENTS

Special thanks are in order for Dr. Picone, Mary Weber, Scott Wilson, Chris Conner, Dr. Follet, Mr. John Vickery, and Elvis.

6. REFERENCES

1. .Murano, Kazuo; Unagami, Shigeyuki; Amano, Fumio; "Echo Cancellation and Applications," in IEEE Communications Magazine, v. 28 pp. 49-55, January '90.
2. Petillon, Thierry; Gilloire, Andre; Theodoridis, Sergios; "The Fast Newton Transversal Filter: An Efficient Scheme for Acoustic Echo Cancellation in Mobile Radio," in IEEE Transactions on Signal Processing, v. 42 pp. 2134-44 May '94.
3. Gilloire, Andre; Vetterli, Martin; "Adaptive Filtering in Subbands with Critical Sampling: Analysis, Experiments, and Application to Acoustic Echo Cancellation," IEEE Transactions on Signal Processing, v. 40 pp. 1862-75, August '92.
4. van de Kerkhof, Leon M.; Kitzen, Wil J. W.; "Tracking of a Time-Varying Acoustic Impulse Response by an Adaptive Filter," in IEEE Transactions on Signal Processing, v. 40 pp. 1285-94, June '92.
5. Clark, Robert; "Active Damping on Enclosed Sound Fields Through Direct Rate Feedback Control," in Acoustical Society of America, pp. 1710-16, March 1995.
6. Gingell, M.J.; Hay, B.G.; Humphrey, L.D. "A Block Mode Update Echo Canceller Using Custom LSI," GLOBECOM Conference Record, v. 3, 1394-97 (November 1983).
7. Kates, James M.; "Feedback Cancellation in Hearing Aids: Results from a Computer Simulation," in IEEE Transactions on Signal Processing, v 39 pp. 553-62, March '91.
8. Kuo, Sen M.; Pan Zhibing; "Distributed Acoustic Echo Cancellation System with Double-Talk Detector," in The Journal of the Acoustical Society of America, v. 94 pp. 3057-60 December '93
9. Messerschmitt, David; Hedberg, David; Cole, Christopher; Haoui, Amine; Winship, Peter; "Digital Voice Echo Canceller with a TMS32020," in Digital Signal Processing Applications with the TMS320 Family, pp. 415-437, Texas Instruments, Inc., 1986.
10. Messerschmitt, David; "Echo Cancellation in Speech and Data Transmission," in IEEE Journal on Selected Topics in Communications, SAC-2, No. 2, 283-303 (March 1984).
11. Picone, J.; Johnson, M.A.; Hartwell, W.T.; "Enhancing Speech Recognition Performance with Echo Cancellation," in Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing, pp. 529-532, New York, New York, USA, April 1988.
12. Ruckman, C.E;, "Active Noise Control FAQ," Available via anonymous ftp as from ftp://rtfm.mit.edu/pub/usenet/news.answers/active_noise_control_faq.
13. Shieh, Chifong; Bai,Mingsian R.; "Active Noise Cancellation by Using The Linear Quadratic Gaussian Independent Modal Space Control," Acoustical Society of America., pp. 2664-2671, May 1995.
14 Sondhi, Man Mohan; Berkley, David A.; "Silencing Echoes on the Telephone Network," in Proceedings of the IEEE, v. 68 pp. 948-63, August 1980.
15. Tokhi, M.O.; Leitch, R.R.; Active Noise Control, Oxford Science Publications, Clarendon Press 1992.
16. Proakis, J.G.;. Manolakis, D.G.; Digital Signal Processing: Principles, Algorithms, and Applications, 2nd Edition, Macmillan, New York, New York, USA, 1992.
17. Schildt, Herbert; Teach Yourself C, 2nd ed., McGraw Hill, Berkeley, California, USA, 1994.
18. The Student Edition of MATLAB, Prentice Hall, Englewood Cliffs, New Jersey, USA, 1992.
19 On the Net/Current DSP Hardware:
20. http//www.ti.com/sc/docs/dsps/softcoop/ Software Overviews for several different echo cancellation algorithms.

7. Appendix A: Echo Canceller Algorithm

The following information is a generic echo cancellation algorithm quoted from [9]. This publication describes a standard transversal filtering algorithm:

“The reflected echo signal r(i) at time i can be written as the convolution of the far-end reference signal y(i) and the discrete representation h_k of the impulse response of the echo path between port C and D.

$$r(i) = \sum_{k=0}^{N-1} h_k y(i-k)$$

Linearity and a finite duration N of the echo-path response have been assumed. An echo canceller with N taps adapts the N coefficients a_k of its transversal filter to produce a replica of the echo r(i) defined as follows:

$$\hat{r}(i) = \sum_{k=0}^{N-1} a_k y(i-k)$$

Clearly, if $a_k = h_k$ for k = 0, 1, ..., N-1, then $\hat{r}(i)$ =r(i) for all time i and the echo is cancelled exactly.

Since, in general, the echo-path impulse response h_k is unknown and may vary slowly with time, a closed-loop coefficient adaptation algorithm is required to minimize the average or mean-squared error (MSE) between the echo and its replica. It can be determined that the near-end error signal u(i) is comprised of the echo-path error r(i) - $\hat{r}(i)$ and the near-end speech signal x(i), which is uncorrelated with the far-end signal y(i). This gives the equation

$$E(u^2(i)) = E(x^2(i)) + E(e^2(i))$$

where E denotes the expectation operator. The echo term $E(e^2(i))$ will be minimized when the left-hand side of (3!!!) is minimized. If there is no near-end speech (x(i) = 0), the minimum is achieved by adjusting the coefficients a_k along the direction of the negative gradient of $E(e^2(i))$ at each step with the update equation

$$a_k(i+1) = a_k(i) - \beta \frac{\partial E(e^2(i))}{\partial a_k(i)}$$

where β is the step size. Substituting (2) and (3) into (4) gives from (5) the update equation

$$a_k(i+1) = a_k(i) + 2\beta E[e(i)y(i-k)]$$

In practice, the expectation operator in the gradient term $2\beta E[e(i)y(i-k)]$ cannot be computed without a priori knowledge of the reference signal probability distribution. Common practice is to use an unbiased estimate of the gradient, which is based on time-averaged correlation error. Thus, replacing the expectation operator of (6) with a short-time average, gives

$$a_k(i+1) = a_k(i) + 2\beta \frac{1}{M} \sum_{m=0}^{M-1} e(i-m)y(i-m-k)$$

he special case of (7) for M=1 is frequently called the least-mean squared (LMS) algorithm or the stochastic gradient algorithm. Alternatively, the coefficients may be updated less frequently with a thinning ratio of up to M, as given in

Computer simulations of this “block update” method [not shown here] show that it performs better than the standard LMS algorithm (i.e. M=1 case) with noise or speech signals[10]. Many cancellers today avoid multiplication for the correlation function in (8), and instead use the signs of e(i) and y(i-k) to compute the coefficient updates. However, this “sign algorithm” approximation results in approximately a 50% decrease in convergence rate and an increase in degradation of residual echo due to interfering near-end speech.

The convergence properties of the algorithm are largely determined by the stepsize parameter β and the power of the far-end signal y(i). In general, making β larger speeds the convergence, while a smaller β reduces the asymptotic cancellation error.

It has been shown that the convergence time constant is inversely proportional to the power of y(i), and that the algorithm will converge very slowly for low-power signals[11]. To remedy that situation, the loop gain is usually normalized by an estimate of that power, i.e.,

$$2\beta = 2\beta(i) = \frac{\beta_1}{P_y(i)}$$

where β_1 is a compromise value of the stepsize constant and $P_y(i)$ is an estimate of the average power of y(i) at time i.

$$P_y(i) = (L_y(i))^2$$

where $L_y(i)$ is given by

$$L_y(i+1) = (1-\rho)L_y(i) + \rho|y(i)|$$

The estimate $\rho y(i)$ is used since the calculation of the exact average power is computation-expensive."