Acoustic Echo Cancellation

Khyle Calpe and Phillip Kwan

***Abstract*—In acoustic echo cancellation, far-end speech that is detected in the near-end speech in the form of echoes is removed by an adaptive finite impulse response (FIR) filter. Voice communication over wireless mobile devices is becoming increasingly widespread, and there is a need to improve audio quality via removing echoes accurately and quickly on embedded systems. This paper discusses the implementation of a normalized least mean square (NLMS) algorithm to update the adaptive filter for fast echo cancellation on an STM32H7 microcontroller (MCU) board while maintaining good audio quality. The implementation removed echoes while preserving the audio quality of the input signal with a processing time mean of 11 seconds for 4 seconds of audio.**

Index Terms*—Acoustic Echo Cancellation (AEC); Adaptive Finite Impulse Response (FIR) Filtering; Geigel (Double-Talk Detection) DTD; far-end, near-end, Least Mean Square (LMS), Normalized LMS (NLMS), Recursive Least Square (RLS)

I. Introduction

A. History

The origins of noise cancellation trace back to two-wire local circuits of conventional analog telephones to local central offices. Due to circuit losses in the case of long-distance communication, mono-directional amplifiers grew in use but needed four-way wire lines connected by devices called hybrids, which needed balanced impedances to prevent signal reflection. Due to the variability of impedance across transmission lines, echo suppressors were invented to minimize echoes. In satellite communications, an alternative was necessary due to long delays in wave propagation [1].

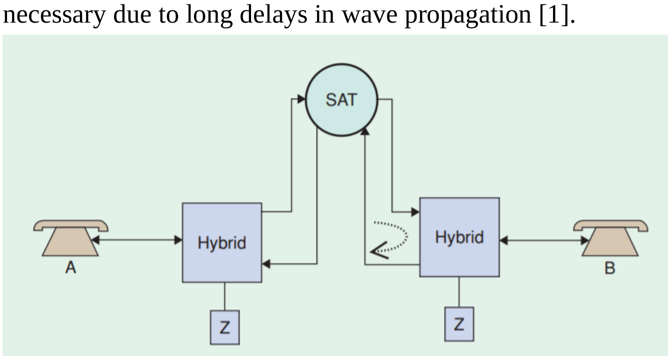


Figure 1. Local circuit of analog telephones.
Courtesy of IEEE [1].

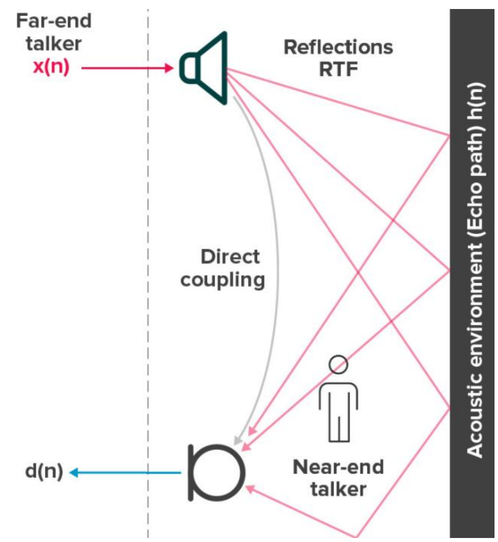


Figure 2. Echo Generation.
Courtesy of audioXpress. [3].

Noise cancellation achieved similar functionality by replicating echoes by filtering far-end signals and subtracting the echoes from the return signal. Due to the non-stationary property of echo paths, an adaptive filter algorithm was proposed by John Kelly, Jr. of Bell Labs [1]. A digital equivalent was also formulated using the LMS algorithm and later the NLMS algorithm, which is still used in current echo cancellers [2].

Echo exists in numerous situations in communication systems such as video-conferencing systems, mobile hands-free phones, videophone terminals, mobile communication, satellite communication, etc. [4] Especially, with the widespread use of wireless mobile communication, the need for better and faster echo cancellation technologies that can be applied on embedded systems arise.

Numerous algorithms have been proposed to address specific applications of echo cancellation and improve upon prior algorithms. Gradient-descent adaptive algorithms such as NLMS and recursive least squares (RLS) perform echo cancellation well with variations in computational complexity and convergence times. Similarly, DTD can be performed by various algorithms such as variable impulse response (VIRE) DTD and cross-correlation DTD with variations in implementation complexity [5].

B. Global Constraints

The project's constraints primarily originate from the peripheral devices used to record and output audio signals. In particular, the durability of the microphone and speaker limits the environment for testing. This was evident when one of each device broke down during testing. Additionally, the cost of the devices translated to the quality of the audio measurement and output devices, which introduced a significant presence of static noise during audio playback. In addition, low amplitude audio signals are also inaudible due to the noise. For projects dealing with audio playbacks, higher quality equipment is necessary despite the increase in cost. Furthermore, the remote nature of the development hindered the speed and progress of the project.

II. Motivation

LMS based algorithms have been the most widely used version of adaptive filtering for noise and echo cancellation due to low computational complexity and stability. However, the convergence rate of the algorithm is too slow for cases where the inputs change fast such as in echo cancellation and does not work well in cases when both the far-end and near-end speech are present. Double talk detection and higher convergence rate algorithms that fix these problems may require too much computational power [5]. As a result, research has been done to develop new learning algorithms such as the NLMS and DTD schemes such as VIRE DTD for faster convergence rates without sacrificing computational power and audio quality [6].

As mobile communication is becoming more popular, echo cancellation algorithms needed to be developed that could both run fast on embedded processors and require minimal memory usage while maintaining or improving audio quality [4]. We wanted to investigate the algorithm efficiency required for good echo cancellation on embedded processors and potentially expanding to real time processing.

III. Approach

The work for implementing the NLMS algorithm for acoustic echo cancellation is split into five components: echo generation, NLMS adaptive filtering, double-talk detection, nonlinear processing, and audio processing.

A. Team Organization

Our group specifically split the work two-way on echo generation and adaptive filter development. Tuning parameters for improving the NLMS and DTD algorithm was done collectively.

B. Plan

Week	Plan	What Happened
3	Implementation of echo generator and FIR filter	Debugging the STM32H7 and code due to incorrect code and memory faults
5	Development and tuning of NLMS adaptive filter and double-talk detection	Finished implementation of both echo generator and FIR filter
7	Development of nonlinear processor and comfort noise generator	-Tuning double talk detection and NLMS algorithm -Some optimizations were made to reduce processing time
10	Completion of acoustic echo cancellation system	-Comfort noise generator was deemed unnecessary due to noise in the speaker -Audio filters were introduced to improve audio quality

Table 1. 10-week plan of the project workload.

C. Standard

Echoes are a major issue that must be addressed in voice communication protocols such as VoIP in order to improve audio quality [7]. Acoustic echo cancellation is commonly implemented using an adaptive FIR filter [8]. While there are many algorithms that can be used to update the FIR filter coefficients, a commonly used one that we have chosen is NLMS due to its relatively fast convergence time and low computation complexity [9].

D. Theory

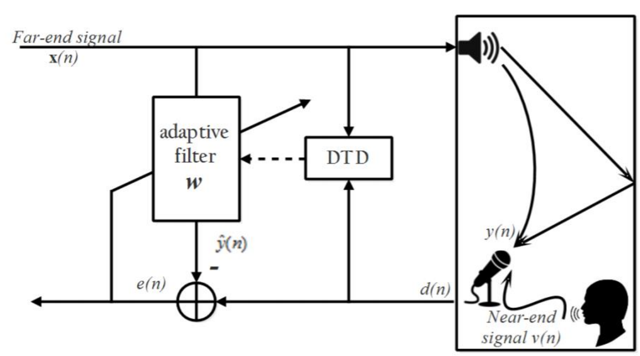


Figure 3. Block Diagram of AEC.
Courtesy of IEEE [10].

The purpose of the adaptive filter in echo cancellation is to identify the room impulse response or echo path **h** that created the echo [8]. A FIR filter is typically chosen due to its stability and low computation complexity in updating the weights **w**[n] [5], where n is the index of the sample. From the far-end signal **x**[n], the echo signal **y**[n] is produced by the following equation:

$$\boldsymbol{y}[\mathrm{n}] = \boldsymbol{h}^{\boldsymbol{T}} \cdot \boldsymbol{x}[\mathrm{n}]$$

, where $\boldsymbol{h} = [h_0, h_1, \ldots, h_{L-1}]$ and L is the length of the echo path and FIR filter.

The microphone signal **d**[n] is the superposition of the near-end signal **v**[n] and the echo signal **y**[n].

$$\boldsymbol{d}[\mathrm{n}] = \boldsymbol{v}[\mathrm{n}] + \boldsymbol{y}[\mathrm{n}]$$

Assuming good convergence, the adaptive filter estimates the echo signal $\widehat{\boldsymbol{y}}$ and subtracts it from the microphone signal to produce the error signal **e**[n], which is also the echo-cancelled microphone signal.

$$\widehat{\boldsymbol{y}}[\mathrm{n}] = \boldsymbol{w}^{\boldsymbol{T}} \cdot \boldsymbol{x}[\mathrm{n}]$$

$$\boldsymbol{e}[\mathrm{n}] = \boldsymbol{d}[\mathrm{n}] - \widehat{\boldsymbol{y}}[\mathrm{n}]$$

The problem then becomes how to update the adaptive filter coefficients in order to converge to the echo path. The algorithm used in our acoustic echo canceller is the NLMS, which updates the filter using a variable step size and offers faster convergence than the LMS [5]. The form of the update algorithm is:

$$\boldsymbol{w}_{n+1}[\mathrm{k}] = \boldsymbol{w}_n[\mathrm{k}] + \frac{\mu_0}{\boldsymbol{x} \cdot \boldsymbol{x} + \sigma} \boldsymbol{e}[\mathrm{n}]\boldsymbol{x}[\mathrm{n} - \mathrm{k}]$$

, where μ_0 is the base step size that is normalized by the signal energy $\boldsymbol{x} \cdot \boldsymbol{x}$ and σ is a small constant to avoid division by small numbers.

To improve the convergence rate of the filter, pre-whitening is applied to both update parameters **x** and **e**. Pre-whitening makes the signals behave more statistically like white noise, flattening the energy spectra and decorrelating the signals [9]. Furthermore, a bandpass filter was placed at the output of the system to remove unwanted distortions and crackling in the audio.

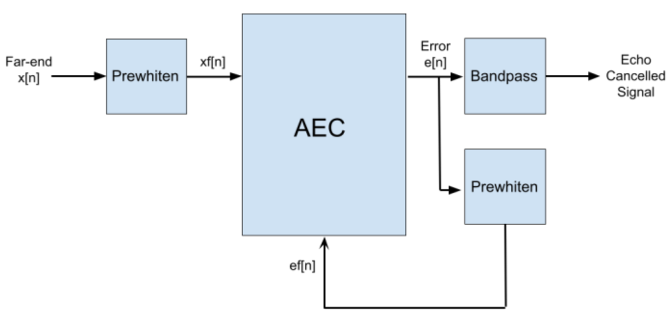


Figure 4. Block Diagram of Audio Processing Subsystem

However, we want the adaptive filter to update only when the echo signal is present. Double talk occurs when the near-end and echo signal are present in the microphone signal, causing the filter to diverge. A simple double talk detector can be implemented using the Geigel algorithm, which compares the levels of the current microphone sample with the max level of the past L samples of the far-end signal. The algorithm is a threshold based model and works with knowledge that the echo path attenuates the echo signal [11].

$$\xi_n = \frac{|\boldsymbol{d}[\mathrm{n}]|}{\max\left(|\boldsymbol{x}[\mathrm{n}]|, |\boldsymbol{x}[\mathrm{n}-1]|, \ldots, |\boldsymbol{x}[\mathrm{L}-1]|\right)}$$

$$\xi_n < Threshold$$

When double talk is declared, the filter coefficients are frozen by a timeout to prevent false positives in double talk detection [7]. If double talk is not detected, then a nonlinear processor attenuates at the output any residual echo or noise that the echo canceler cannot get rid of.

E. Software / Hardware

This project was built using the STM32H7 board using the Cortex M7 MCU. This microcontroller was chosen for its clock speeds and dedicated floating point unit. Development of the project took place in CubeIDE. To take full advantage of the hardware acceleration, we imported the Arm CMSIS DSP library. All code used in the project is written in C.

F. System Build / Operation

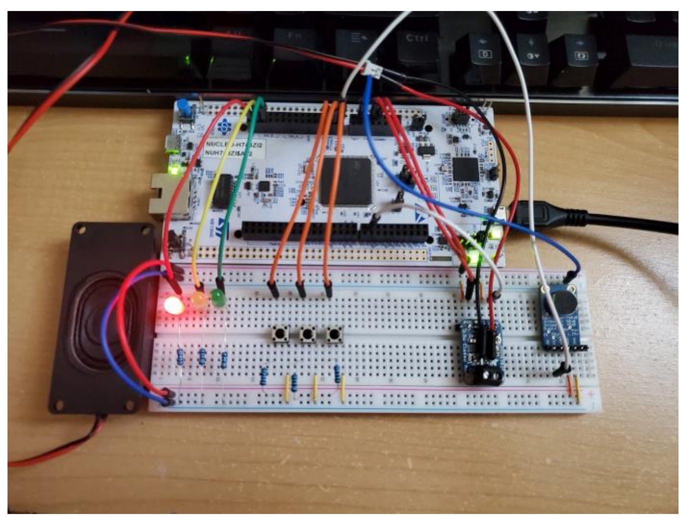


Figure 5. Hardware Setup of Echo Canceler

The circuit of the system utilizes an STM32H7 MCU board, a microphone for the audio recording, an amplifier connected to a speaker for audio playback, three buttons and LEDs for triggering various phases of the AEC process.

To prevent adaptive algorithm divergence, the system uses a Geigel algorithm for double-talk detection. When double-talk is not detected, a non-linear processor removes residual echoes in the signal. A pre-whitening filter is applied to the update parameters of the adaptive filter to improve convergence time by flattening the energy spectra and decorrelating the signals. For further improvement of audio quality, a bandpass filter with passband from 300 to 4000 Hz is applied to the output.

The operation of the AEC proceeds in three steps. The first phase records an audio sample at 8 kHz for 4 seconds. The second phase produces an echo effect by superimposing the signal with a randomly delayed and attenuated version of itself. In the last phase, the processed audio signal is fed into an adaptive FIR filter with an NLMS algorithm and outputted through the speaker.

IV. Results

The acoustic echo canceler runs on an embedded system while maintaining audio quality of the source signal.

A. *Description of Results*

The system successfully processed a 4-second audio sample in approximately 11 seconds while preserving the audio quality of the source. This is a vast improvement of initial trials which processed 2-second audio in approximately 43 seconds. The system accomplishes this by utilizing various optimization methods.

B. *Discussion of results*

The reduction of the processing time for a longer audio sample was difficult to implement in the STM32H7 board. The issues surfaced due to the memory constraints of the board, which limited the memory size of the audio buffer and the sampling rate. In the case of the memory, the board fails to execute due to the program's memory requirements exceeding the onboard RAM size of the board. Furthermore, filter length directly impacts the processing time. Longer filters remove echoes better but take longer to run. Thus, we needed to balance audio quality without sacrificing too much processing time.

To increase execution time while preserving audio quality, our group implemented memory access optimizations, the C library's `memmove` and `memset` functions, and the ARM CMSIS DSP library functions. To further increase execution time, the STM32H7's clock frequency could be increased. Processing time can be reduced by increasing the clock frequency of the MCU or using alternative learning algorithms such as RLS that may require more computational power but the increased convergence rate enabling reduction in filter length may outweigh the downsides.

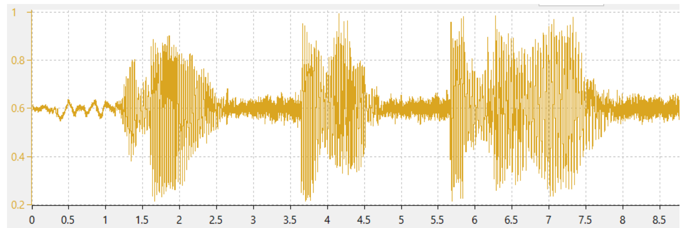


Figure 6. Normalized time signal of the words "Big-Red-Demo."

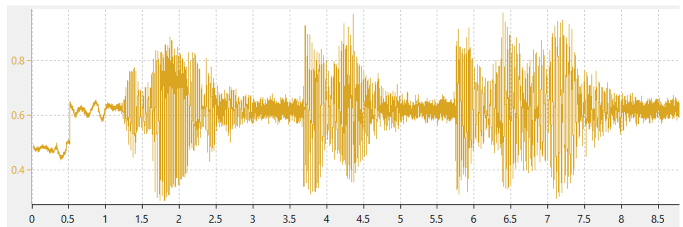
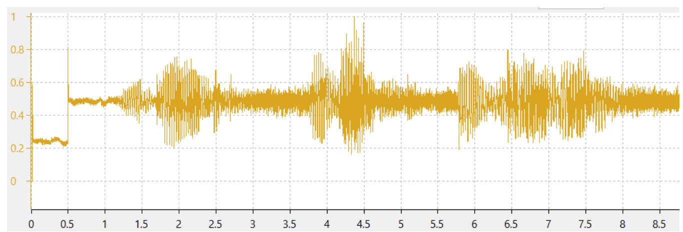


Figure 7. Time signal with an echo.



Figures 8. Post-processed time signal with the adaptive filter.

The system preserved the audio quality from the audio input to the output with minimal distortion. In the following figures, the audio signals indicate that the filter fails to cancel out echoes during the beginning of the audio signal. This is a result of the filter's learning process, which only converges during the latter stages of the audio signal. Recordings with longer samples can potentially improve the quality by increasing the rate of convergence. Furthermore, better DTD algorithms such as VIRE can greatly reduce false positives and improve the quality of the echo cancellation.

V. Demonstration

Below is a link to a video of our project running.

https://youtu.be/jFF2QquR8xs

References

[1] Man Mohan Sondhi, "The history of echo cancellation," in IEEE Signal Processing Magazine, vol. 23, no. 5, pp. 95-102, Sept. 2006, doi: 10.1109/MSP.2006.1708416.

[2] B. Widrow, "Thinking about thinking: the discovery of the LMS algorithm," in IEEE Signal Processing Magazine, vol. 22, no. 1, pp. 100-106, Jan. 2005, doi: 10.1109/MSP.2005.1407720.

[3] E. Yorga, *R&d stories: Acoustic echo canceller calibration in real time*. 17-June-2020. [Online]. Available: https://audioxpress.com/article/r-d-stories-acoustic-echo-canceller-calibration-in-real-time [Accessed: 12-Mar-2021].

[4] Jun, Sun & Bin, Dai & Shou-Yong, Zhang. (2011). A New Algorithm of Echo Cancellation in Mobile Applications. 232. 443-449. doi: 10.1007/978-3-642-23998-4_61.

[5] A. Jovanovic, K. Nilvér, P. Söderberg, & M. Broberg, (2006). "Acoustic Echo Cancellation," *Uppsala University*. http://www2.ece.rochester.edu/courses/ECE472/resources/Papers/Uppsala_2006.pdf

[6] J. Dhiman, S. Ahmad, and K. Gulia, "A Review of Adaptive Line Enhancers for Noise Cancellation," *International Journal of Science, Engineering and Technology Research (IJSETR)*, vol. 2, no. 5, May 2013.

[7] M. S. Hussain, M. A. Hasan, M. F. Bari and A. B. M. Harun-Ur-Rashid, "A fast double-talk detection algorithm based on signal envelopes for implementation of acoustic echo cancellation in embedded systems," *2017 4th International Conference on Advances in Electrical Engineering (ICAEE)*, Dhaka, Bangladesh, 2017, pp. 199-204, doi: 10.1109/ICAEE.2017.8255353.

[8] M. Hamidia and A. Amrouche, "Improving Acoustic Echo Cancellation in Hands-free Communication Systems," 2019 6th International Conference on Image and Signal Processing and their Applications (ISPA), Mostaganem, Algeria, 2019, pp. 1-5, doi: 10.1109/ISPA48434.2019.8966823.

[9] H. Besbes and S. Ben Jebara, "The pre-whitened NLMS: a promising algorithm for acoustic echo cancellation," 9th International Conference on Electronics, Circuits and Systems, Dubrovnik, Croatia, 2002, pp. 875-878 vol.3, doi: 10.1109/ICECS.2002.1046387.

[10] Broadcom, *Addressing Echo in VoIP Systems*, 16-Jul-2019. [Online]. Available: https://techdocs.broadcom.com/us/en/ca-enterprise-software/it-operations-management/unified-communications-monitor/4-3-1/troubleshooting/addressing-echo-in-voip-systems.html.[Accessed: 12-Mar-2021].

[11] M. Benziane, M. Bouamar and M. Makdir, "Double-Talk Detection Based on Enhanced Geigel Algorithm for Acoustic Echo Cancellation," 2018 6th International Conference on Control Engineering & Information Technology (CEIT), Istanbul, Turkey, 2018, pp. 1-5, doi: 10.1109/CEIT.2018.8751867.