BI-MAGNITUDE PROCESSING FRAMEWORK FOR NONLINEAR ACOUSTIC ECHO CANCELLATION ON ANDROID DEVICES

Yiteng (Arden) Huang, Jan Skoglund, Alejandro Luebs

Google Inc., USA
{ardenhuang, jks, aluebs }@google.com

ABSTRACT

This paper presents a new paradigm for acoustic echo control on mobile Android devices. The echo path on these devices has nonlinearities including not only the results of overdriven power amplifiers and miniaturized loudspeakers, but also those caused by hardware audio dynamic range compressor (ADRC). While the former form of nonlinearities was widely investigated in past research, the latter has not yet been taken into account. The ADRC adds extra gains to the echo path and makes it become a fast time-varying system. This presents a great challenge to traditional (both linear and nonlinear) echo cancellation systems. Here we propose a novel bi-magnitude processing framework, which is based on a two-state model for the echo path. The algorithm can deal with the ADRC problem and offers robust control for identification of input nonlinearities. The performance of the proposed approach is evaluated on recordings made in an anechoic chamber using real Android devices.

Index Terms— Nonlinear acoustic echo cancellation, audio dynamic range compressor, bi-magnitude processing framework, branch switching, robust control

1. INTRODUCTION

In two-way or multi-party voice communication, acoustic echo occurs due to the coupling between loudspeakers and microphones. Strong and long-delayed acoustic echoes are annoying and can sometimes completely disrupt a conversation. Acoustic echo cancellation and suppression (AEC and AES) are methods to prevent echoes from being heard. They are enabling technologies for Voice over Internet Protocol (VoIP). AEC achieves the goal by first identifying the echo path using an adaptive filter, then synthesizing a replica of the echo, and finally subtracting it out from the returned signals. Consequently the conversation remains full duplex. AES works by adjusting the gain of local microphones according to the relative strength of far-end and near-end speech. An aggressive AES can create an experience similar to that of talking on a walkie-talkie, which is essentially half duplex. So AEC causes less distortion to local speech than AES. However, AES is computationally less demanding and can offer more echo attenuation than AEC. A sophisticated VoIP platform shall implement both AEC and AES while a less powerful device may consider only an AES solution.

WebRTC is a disruptive technology with the aim of bringing real-time communications (RTC) to web browsers and mobile applications. It was first released by Google in 2011 as an open-source project and has been followed by ongoing work to standardize the relevant protocols in the IETF and browser APIs in the W3C [1]. Recently WebRTC started to pick up steam: as of 2016, there have been more than 1.5 billion WebRTC-enabled browsers and more than 3.0 billion WebRTC mobile app downloads. But meanwhile it has been realized that many components need to be optimized in order to fulfill the great potential of WebRTC. They include AEC and AES.

Today, WebRTC has two software (SW) AEC modules: one for regular desktops and laptops (referred to simply as AEC) and the other for mobile devices (called AEC Mobile or AECM for short). The AEC involves both a linear canceler and a suppressor while the AECM is purely a suppressor. Mobile devices often have hardware (HW) AECs but it has been found that their performance is not always satisfactory. The HW-AEC can even stop working on specific devices. If that happens, the only way to get it work again is to do a factory reset of the device. This rather unpleasant experience drives us to look for the option of running a well-performing SW-AEC on mobile devices for WebRTC.

The existing WebRTC SW-AEC follows the traditional AEC theory and assumes a *linear* echo path [2, 3]. This is quite true for audio subsystems on desktops and laptops. But it does not hold well for typical mobile devices, where small size, low power, low quality, and cost effective amplifiers, loudspeakers, and microphones may be used. These imperfect elements will introduce nonlinear echoes with which the regular SW-AEC cannot deal, leading to significant performance degradation in removing echoes.

In this work, we are set to investigate the problem of nonlinear AEC (NAEC) on mobile phones, in particular Android devices. There was a very rich literature on NAEC in the last two decades [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15]. A wide variety of sources have been considered, ranging from overdriven power amplifiers to small loudspeakers and to saturated microphones. But we found in our own practice that the so-called hardware audio dynamic range compressor (HW-ADRC) could induce a dominant form of nonlinear distortion on Android platforms. Such a form of nonlinearity leads to a quick variation of gain in the echo path, which presents a great challenge for both traditional linear and nonlinear AECs.

Obviously the best and easiest solution for ADRC-induced nonlinear echo problem could be to turn off the HW-ADRC and develop a software replacement for WebRTC. But this does not seem to be a feasible option. It is not clear whether HW-ADRCs can be turned off universally on all Android devices. This situation drove us to develop a more robust AEC framework that can deal with the ADRC effect. In this paper, we propose a bi-magnitude processing algorithm as a fruit of our research effort.

2. CHARACTERIZING MOBILE PHONE AUDIO SYSTEMS

Figure 1 depicts the functional block diagram of a typical monophonic audio subsystem on Android devices. It shows that the echo path is composed of the following audio elements: digital-to-analog converter (DAC), power amplifier, HW-ADRC, loudspeaker,

978-1-5090-2007-2/16/$31.00

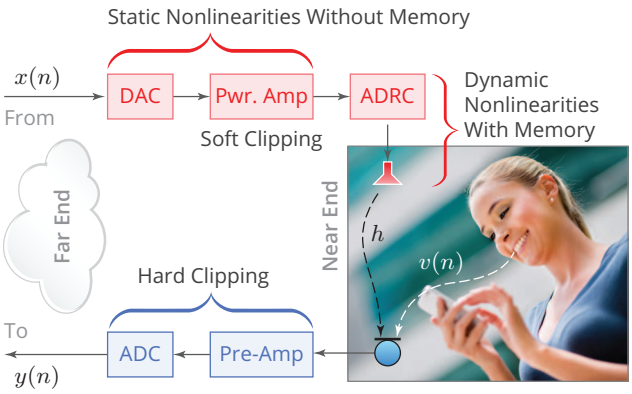

Fig. 1: Block diagram of a typical monophonic audio subsystem on Android devices. Note that ADRC stands for audio dynamic range compressor.

Fig. 2: AppRTC testbed for AEC data collection with Android devices.

surrounding sound field, microphone, pre-amplifier, and analog-to-digital converter (ADC).

Oftentimes mobile phone users want to have high sound levels over distortion. So the power amplifier is commonly overloaded even when only low battery voltage is available. This leads to nonlinear distortion with a characteristic of saturation, which can be described by a soft-clipping function [4]. This form of nonlinearity is typically static and thus can be approximated by a truncated Taylor series expansion. Empirically a fifth-order polynomial can sufficiently reproduce the effect of soft-clipping saturation.

According to [16], nonlinearities of electrodynamic loudspeakers are attributable to three non-ideal parts: acoustical, electromagnetic, and mechanical parts. Generally speaking, it is difficult to precisely model all these nonlinearities. A common approach is to approximate the nonlinear behavior of loudspeakers by a Volterra filter. But loudspeakers on mobile phones have small form factors. For miniaturized loudspeakers, a short memory support for their nonlinearities can be expected [16]. There are many papers suggesting to use only power filters to model loudspeaker nonlinearities; please see [7, 17, 9, 15] and references therein.

While lately there have been several high-end smartphones to feature 24-bit audio for both play-back and recording, most of today's mobile phones still can only handle 16-bit digital sounds. Moreover mobile phones can be used in any acoustic environments and some may have very high sound pressure levels (SPLs). This implies that we don't have a lot of headroom to avoid digital hard clipping. Since the loudspeaker is very close to the microphone on mobile phones, if gain staging is improperly managed [18, 19], hard-clipping distortion can become even more prominent. Hard clipping is best described by a piecewise linear function, which is not differentiable at the clipping points. This can cause difficulty in developing adaptive identification algorithms as required for AEC.

It was explained in the introduction that the aforementioned sources of nonlinearities have been extensively studied in the past. But the ADRC effect on AEC, to the best of our knowledge, has not yet been addressed before. An ADRC is designed to reduce or narrow down an audio signal's dynamic range [20]. There are two methods of compression: downward compression reduces loud sounds over a certain threshold while quiet sounds remain unaffected; upward compression increases the loudness of sounds below a certain threshold while leaving louder sounds unaffected. Many ADRCs provide adjustable control over how quickly they act. They also offer options of whether the bend in the response curve between below threshold and above threshold is abrupt (hard knee) or gradual (soft knee). Some ADRCs may respond to the instantaneous level of the input signal (peak sensing) while other apply an averaging function (commonly RMS) on the input signal before comparing its level to the threshold (RMS sensing). The ADRC effect is dynamic and the nonlinear distortion that it induces is difficult to model.

3. DATA COLLECTION AND PROBLEM IDENTIFICATION

To study nonlinear echoes on Android devices, we used AppRTC as the testbed to collect audio data as shown in Fig. 2. AppRTC is Google's open source testing application for WebRTC. When you go to the AppRTC website [21] using a WebRTC-enabled browser (e.g., Google Chrome, Opera, or Mozilla Firefox), a new video conferencing room is automatically created and a URL is provided for you to share with somebody else. When other people use the URL to join into the virtual room, you connect with them and can start to talk. The AppRTC allows to export the far-end and microphone signals into a binary dump file from which audio data can be extracted offline. It is noteworthy that the AppRTC has been hacked a little bit to turn off hardware AEC, automatic gain control (AGC), and noise suppressor (NS). AppRTC's source code can be downloaded from Github using the link listed in [22].

Data collection was carried out in an anechoic chamber. An acoustically well-controlled environment minimizes irrelevant sources of variation in data analysis and algorithm development. It leads to a relatively fixed linear impulse response in the collected data, allowing us to focus on the nonlinear characteristics of the echo path. During data collection, the slow A-weighted SPL was as low as 10.9 dBA.

We designed 10 wave files for the far-end signals: 6 clean speech including 3 female and 3 male speakers, a 1 kHz sinusoid, a mixture of 1 kHz and 2.3 kHz sinusoids, a frequency sweep signal, and white noise. Four Nexus devices were used: 5x, 6, 6p, and 7. In each case, data were collected for three different play-out volumes: low, moderate, and high. Each recording lasts for about 2 minutes. Most parts of the recordings have only echoes but each recording has a short period of double talks with a duration of less than 20 s.

When a mixture of probing sinusoids is played from the far end, strong intermodulation distortion (IMD) appears in the microphone output on Nexus 5x. This infers the existence of static audio nonlin-

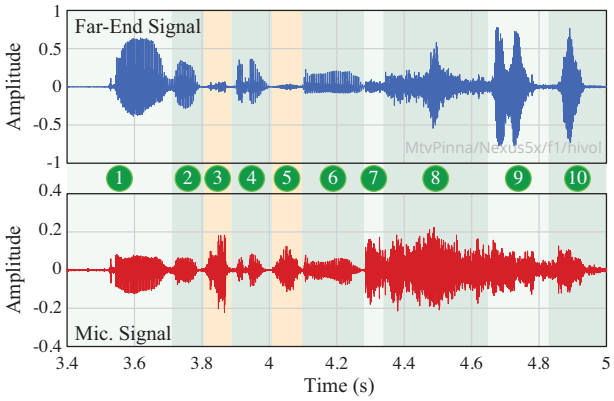

Fig. 3: Demonstration of the influence of hardware audio dynamic range compressor on the gain of the echo path on Nexus 5x.

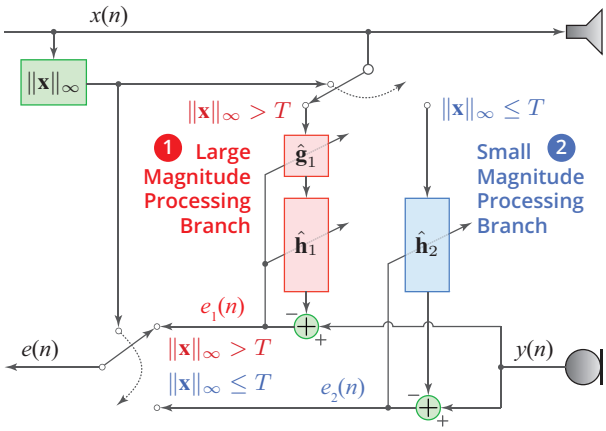

Fig. 4: Illustration of the bi-magnitude processing algorithm to handle the nonlinearity introduced by hardware ADRC on Android devices.

earities in the system. But the most prominent difference between desktop/laptop and Nexus 5x AEC data is that when the far-end signal is speech, the microphone output of a Nexus 5x shows the effect of ADRC. In the example given by Fig. 3, the highlighted third and fifth phonemes are much weaker than the other phonemes in the far-end signal but become as strong as the others in the microphone output. It looks like that an upward compression method is implemented in the ADRC on Nexus 5x. Moreover the ADRC can add a gain to the echo path for a period of time as short as 100 ms. The resulting gain variation will present a great challenge for traditional AEC (both linear and nonlinear) approaches.

4. BI-MAGNITUDE PROCESSING ALGORITHM

The ADRC makes the echo path change quickly in time. Even when the linear surrounding sound field is stationary, the AEC's adaptive algorithm cannot fully converge and may swing back and forth between two impulse responses. So the AEC performance will be severely degraded.

For a fast time-varying system, the state-space echo path model that the Kalman filter uses to solve the NAEC problem seems to be more powerful than a deterministic model that traditional AEC adaptive algorithms assume. The Kalman filter has many well-known advantages including fast convergence, inherent step-size control, and small errors. But it also has the bad reputation of high computational load and poor numerical stability in the case of long filters [23]. So the focus was not placed on this class of technology. Instead we proposed a novel bi-magnitude processing (BMP) algorithm to deal with this problem as illustrated by Fig. 4.

The BMP algorithm is a deterministic method which has two branches. Each branch has an independent set of adaptive filters. The AEC system switches between the two branches according to the magnitude of the far-end signals. The switching algorithm senses and monitors the peak value of the far-end signal over a sliding window. If it exceeds a preset threshold T measured in dB with respect to the full scale range (FSR) of input signal, the AEC system works on the first branch by updating its adaptive filters and taking its output as the system's output. When the peak value falls below the threshold, the second branch takes over the AEC system. We call the first branch the large magnitude processing (LMP) branch and the second branch the small magnitude processing (SMP) branch. This implies that we have assumed two states for the echo path. The state transition depends on the infinite norm of a sliding window of input signal. When the threshold and the window length are properly selected, the two-state model can effectively help characterize the behavior of an ADRC. Branch switching can cause discontinuities in the estimated echo signal. Smoothing those data windows where state transition takes place will reduce artifacts in the AEC output.

In addition to the ability of acting quickly to gain variations in the echo path caused by a HW-ADRC, the BMP framework has another advantage. To recall, audio nonlinearities are usually associated with large magnitudes of input signal. So when the far-end signals are pretty weak, one may not be able to measure any nonlinear distortion even on a very poor audio system. For example, if a low-quality power amplifier is not overloaded and stays strictly away from its saturation zones, then its nonlinearity would not emerge and does not affect AEC. From a system identification perspective, the input nonlinearity is not fully excited and it is not identifiable. In this case, even the most advanced NAEC algorithm will give a spurious estimate of the true nonlinearity. Ideally this spurious estimate is simply a linear function. But usually it has no similarity with the true nonlinearity beyond the range of input values. Since speech in nature has a large dynamic range, the input nonlinearities of an Android device (if any) may sporadically appear. If all the inputs and outputs are indiscriminately presented to an adaptive NAEC algorithm, the algorithm is expected to diverge. This is analogous to the impact of double talks on traditional linear AEC algorithms. The BMP algorithm provides a more robust control for NAEC. As seen in Fig. 4, we include a nonlinear model (a power filter $\hat{\mathbf{g}}_1$) in the LMP branch but leave the SMP branch purely linear.

It needs to be pointed out that due to the limitation of space we present only one possible design in Fig. 4. If the HW-ADRC is existent and the following loudspeaker has severe nonlinearity, faint far-end signals may be amplified to the level that could have been able to trigger the loudspeaker's nonlinearity and it is desirable to incorporate nonlinear models on both branches.

It is noteworthy that while the BMP structure looks similar to that of the dynamic impulse response (DIR) model proposed in [24], the two algorithms are fundamentally different: they deal with two different sources of nonlinearities. The DIR model, which is a piecewise linear approximation of a nonlinear function, needs to trade off between modeling accuracy and adaptation speed [25]. But the BMP system addresses binary gated gains added by an ADRC and does not suffer from the trade-off dilemma.

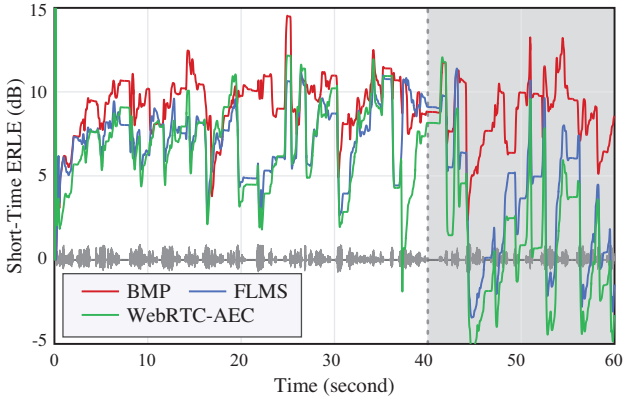

Fig. 5: Performance comparison in ERLE between the BMP, the FLMS, and the WebRTC-AEC.

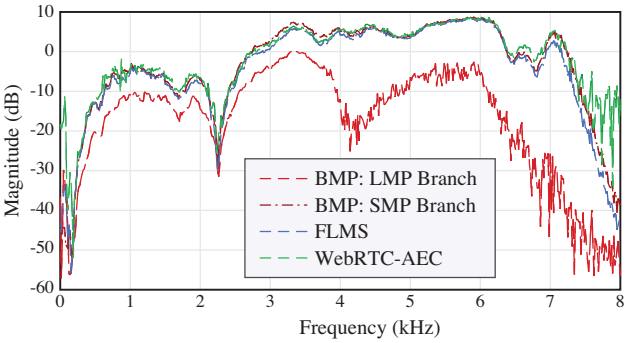

Fig. 6: Frequency responses estimated the BMP (two branches, respectively), the FLMS, and the WebRTC-AEC.

5. EXPERIMENTAL RESULTS

To validate the proposed BMP algorithm, we performed a series of experiments on our AEC data recorded from a Nexus 5x. We compared its performance with that of the linear frequency-domain LMS (FLMS) algorithm [3] and the WebRTC-AEC (canceler only). The sampling rate is 16 kHz. The linear filter length is 768 for all the algorithms. The block shift for the FLMS is 8 ms while the partition size for the WebRTC-AEC is 4 ms. For fair comparison, the BMP results that we presented in this paper are only those that both branches have only linear filters. These filters are estimated by the same FLMS algorithm.

We found in some of our preliminary experiments that when we included a nonlinear power filter $\hat{\mathbf{g}}_1$ in the LMP branch as suggested by Fig. 4, the BMP performance got further improved when the LMP branch was in charge of the system. But the performance gain depended on the NAEC algorithm we chose and on the amount of time during which the far-end signal met the LMP condition. We will address these nonlinear components later.

The recording made for the first male speaker and for the highest possible play-out volume on Nexus 5x is used here as an example. Other recordings produce similar results. For this study, we extract samples over a period of 1 min where there is no local speech. For the first 40 s, the investigated adaptive algorithms adapt as they are supposed to do. But we freeze them afterwards, which simulates the case of double talks. Sometimes the performance of a canceler when its adaptive algorithm is frozen during double talks is more important than that during regular adaptation. In the case of single far-end

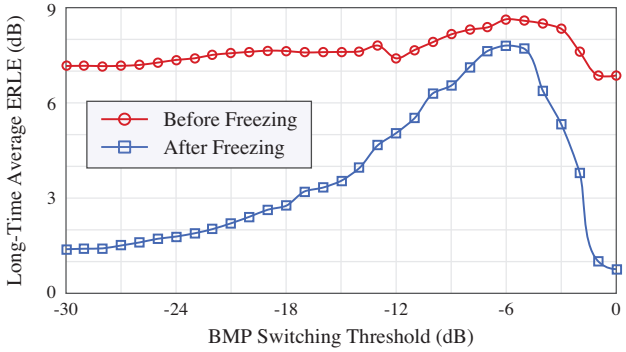

Fig. 7: Effects of the value of the switching threshold (T) on the performance of the BMP algorithm. The threshold is measured in dB with respect to the full-scale range (FSR) of input signal.

talks, the canceler's residual echo can be aggressively attenuated by the following suppressor. But during double talks, aggressive attenuation can impose a strong distortion on local speech and reduce audio transparency of near end. So a good performing canceler is more helpful and hence more desirable.

The performance metric of interest for real-recorded AEC data is the echo return loss enhancement (ERLE) in dB defined as

$$\text{ERLE} \triangleq 10\log_{10}\left\langle y^2(n)\right\rangle_t / \left\langle e^2(n)\right\rangle_t , \tag{1}$$

where $\langle\cdot\rangle_t$ denotes the average of data over a period of t.

Figure 5 presents the short-time ERLE of the three studied algorithms. For short-time ERLE, the analysis window is 200 ms long and shifts every 10 ms. For the BMP, T is set to -6 dB. We see that the BMP clearly outperforms the FLMS and the WebRTC-AEC, particularly after adaptation is frozen as indicated by the gray area. In Fig. 6, we intend to check the impulse responses that those adaptive algorithms estimated. The transfer functions of those impulse responses saved at the end of adaptation are plotted. We see that the impulse responses estimated by the two branches of the BMP differ clearly by a gain. The estimates of the FLMS and WebRTC are close to that estimated by the SMP branch of the BMP algorithm. This probably can explain why the FLMS and WebRTC perform poorly after adaptation is frozen.

Finally we present a study investigating how to choose T in the BMP. We vary T from 0 dB to -30 dB and check the long-time average ERLE before and after adaptation freezing (the average time $t = 40$ s and 20 s, respectively). Figure 7 plots the results. When $T = 0$ and as $T \to -\infty$, only one branch (the SMP branch and LMP branch, respectively) is actually used and then the BMP is identical to the FLMS algorithm. We see that when $T = -6$ dB the BMP achieves the best performance.

6. CONCLUSIONS

This paper studied the nonlinear echo problem caused by hardware audio dynamic range compressor (HW-ADRC) on Android devices. The HW-ADRC adds extra gain to faint far-end signals and hence makes the echo path a fast time-varying system, which forms a great challenge for both traditional linear and nonlinear acoustic echo cancellation algorithms. We proposed a novel bi-magnitude processing algorithm that can effectively deal with this problem. It also offers a robust control for identification of input nonlinearities. The performance of the proposed algorithm was evaluated on recordings made in an anechoic chamber with real Android devices.

7. REFERENCES

[1] Wikipedia (The Free Encyclopedia), "WebRTC," https://en.wikipedia.org/wiki/WebRTC, 2016, [Online; Latest Accessed 9-May-2016].

[2] C. Breining, P. Dreiseitel, E. Hänsler, A. Mader, B. Nitsch, H. Puder, T. Schertler, G. Schmidt, and J. Tilp, "Acoustic echo control – an application of very-high-order adaptive filters," *IEEE Signal Process. Mag.*, vol. 16, no. 4, pp. 42–69, July 1999.

[3] J. Benesty, T. Gänsler, D. R. Morgan, M. M. Sondhi, and S. L. Gay, *Advances in Network and Acoustic Echo Cancellation*, Springer-Verlag, Berlin, Germany, 2001.

[4] B. S. Nollett and D. L. Jones, "Nonlinear echo cancellation for hands-free speakerphones," in *Proc. NSIP*, 1997.

[5] A. Stenger, L. Trautmann, and R. Rabenstein, "Nonlinear acoustic echo cancellation with 2nd order adaptive Volterra filters," in *Proc. IEEE ICASSP*, 1999, vol. 2, pp. 877–880.

[6] A. Stenger and W. Kellermann, "Adaptation of a memoryless preprocessor for nonlinear acoustic echo cancelling," *Signal Processing (Elsevier)*, vol. 80, no. 9, pp. 1747–1760, Sept. 2000.

[7] F. Kuech, A. Mitnacht, and W. Kellermann, "Nonlinear acoustic echo cancellation using adaptive orthogonalized power filters," in *Proc. IEEE ICASSP*, 2005, vol. 3, pp. 105–108.

[8] F. Kuech and W. Kellermann, "Nonlinear residual echo suppression using a power filter model of the acoustic echo path," in *Proc. IEEE ICASSP*, 2007, vol. 1, pp. 73–76.

[9] K. Shi, X. Ma, and G. Zhou, "A channel shortening approach for nonlinear acoustic echo cancellation," in *Proc. IEEE Workshop on Statistical Signal Processing*, 2007, pp. 351–354.

[10] D. A. Bendersky, J. W. Stokes, and H. S. Malvar, "Nonlinear residual acoustic echo suppression for high levels of harmonic distortion," in *Proc. IEEE ICASSP*, 2008, pp. 261–264.

[11] M. I. Mossi, N. W. D. Evans, and C. Beaugeant, "An assessment of linear adaptive filter performance with nonlinear distortions," in *Proc. IEEE ICASSP*, 2010, pp. 313–316.

[12] M. I. Mossi, C. Yemdji, N. W. D. Evans, C. Beaugeant, and P. Degry, "Robust and low-cost cascaded non-linear acoustic echo cancellation," in *Proc. IEEE ICASSP*, 2011, pp. 89–92.

[13] J. M. Gil-Cacho, T. van Waterschoot, M. Moonen, and S. H. Jensen, "Nonlinear acoustic echo cancellation based on a parallel-cascade kernel affine projection algorithm," in *Proc. IEEE ICASSP*, 2012, vol. 1, pp. 33–36.

[14] S. Malik and G. Enzner, "A variational bayesian learning approach for nonlinear acoustic echo control," *IEEE Trans. Signal Process.*, vol. 61, no. 23, pp. 5853–5867, Dec. 2013.

[15] M. Z. Ikram, "Non-linear acoustic echo cancellation using cascaded Kalman filtering," in *Proc. IEEE ICASSP*, 2014, pp. 1320–1324.

[16] W. Klippel, "Dynamic measurement and interpretation of the nonlinear parameters of electrodynamic loudspeakers," *J. Audio Eng. Soc.*, vol. 44, no. 9, pp. 2195–2208, Sept. 1996.

[17] G. Enzner, H. Buchner, A. Favrot, and F. Kuech, "Acoustic echo control," in *Academic Press Library in Signal Processing*, R. Chellappa and S. Theodoridis, Eds., chapter 30, pp. 807–878. Elsevier Academic Press, Chennai, India, 2014.

[18] Wikipedia (The Free Encyclopedia), "Gain stage," https://en.wikipedia.org/wiki/Gain_stage, 2015, [Online; Latest Accessed 2-May-2016].

[19] J. Lewis, "Understanding microphone sensitivity," http://www.analog.com/library/analogDialogue/archives/46-05/understanding_microphone_sensitivity.pdf, 2012, [Online; Latest Accessed 2-May-2016].

[20] Wikipedia (The Free Encyclopedia), "Dynamic range compression," https://en.wikipedia.org/wiki/Dynamic_range_compression, 2016, [Online; Latest Accessed 9-May-2016].

[21] Google Inc., "AppRTC," https://appr.tc, 2016, [Online; Latest Accessed 12-Dec-2016].

[22] Google Inc., "AppRTC Demo Code," https://github.com/webrtc/apprtc, 2016, [Online; Latest Accessed 15-Dec-2016].

[23] S. Haykin, *Adaptive Filter Theory*, Prentice-Hall, Upper Saddle River, NJ, 4 edition, 2002.

[24] S. Saito, A. Nakagawa, and Y. Haneda, "Dynamic impulse response model for nonlinear acoustic system and its application to acoustic echo canceller," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2009, pp. 201–204.

[25] C. Hofmann, C. Huemmer, and W. Kellermann, "Significance-aware Hammerstein group models for nonlinear acoustic echo cancellation," in *Proc. IEEE ICASSP*, 2014, pp. 5975–5979.