Least Squares Signal Declipping for Robust Speech Recognition

Mark J. Harvilla and Richard M. Stern

Department of Electrical and Computer Engineering
Carnegie Mellon University, Pittsburgh, PA 15213 USA
mharvill@cs.cmu.edu, rms@cs.cmu.edu

Abstract

This paper introduces a novel declipping algorithm based on constrained least-squares minimization. Digital speech signals are often sampled at 16 kHz and classic declipping algorithms fail to accurately reconstruct the signal at this sampling rate due to the scarcity of reliable samples after clipping. The Constrained Blind Amplitude Reconstruction algorithm interpolates missing data points such that the resulting function is smooth while ensuring the inferred data fall in a legitimate range. The inclusion of explicit constraints helps to guide an accurate interpolation. Evaluation of declipping performance is based on automatic speech recognition word error rate and Constrained Blind Amplitude Reconstruction is shown to outperform the current state-of-the-art declipping technology under a variety of conditions. Declipping performance in additive noise is also considered.

Index Terms: nonlinear distortion, declipping, robust speech recognition, speech enhancement, constrained optimization

1. Introduction

Signal clipping is a common form of dynamic range compression (DRC) in which the peaks of a signal exceeding a certain amplitude threshold are lost, or *clipped*. Clipping is a non-invertible, many-to-one mapping. Clipping typically occurs in one of three ways: (1) during signal recording, as a result of exceeding the dynamic range limitations of an analog-to-digital (A/D) converter (e.g., by yelling loudly into a microphone and not properly adjusting the pre-amplifier gain), (2) as a result of writing audio data to a file, where the audio has not been properly normalized in amplitude (e.g., MATLAB's `wavwrite` function requires values in the range $[-1, 1]$), or (3) on purpose, to achieve some desirable perceptual characteristic and/or reduce the signal's dynamic range (e.g., for mastering music).

In this paper, clipping is simulated using the following transformation.

$$x_c[n] = \begin{cases} x[n] & \text{if } |x[n]| < \tau \\ \tau \cdot \operatorname{sgn} x[n] & \text{if } |x[n]| \geq \tau \end{cases} \tag{1}$$

In Equation 1, $x[n]$ is an unclipped waveform, and $x_c[n]$ is the clipped output. Clipping is parameterized by the clipping threshold, τ, which will be expressed in terms of percentiles of the absolute value of the unclipped speech, $x[n]$. Expressing τ in this way causes the clipping transformation to be independent of the scaling of the input waveform, and allows for a more controlled experiment. The X^{th} percentile will be denoted P_X. Note that if $\tau = P_X$ then $(1 - P_X)\%$ of the signal samples have been lost due to clipping.

Figure 1 depicts one pitch period of voiced speech, sampled at 16 kHz, that has been clipped at an amplitude of $\tau = P_{75}$, in this case corresponding to 0.0631. The original unclipped speech is shown in grey. The clipped signal is overlaid as a stem plot; the black samples are clipped, the blue samples remain reliable after clipping. There is a clear scarcity of reliable samples in the regions where the original signal has a large amplitude. An ideal declipping algorithm will recover the original (gray) unclipped signal from the observed (black and blue) samples.

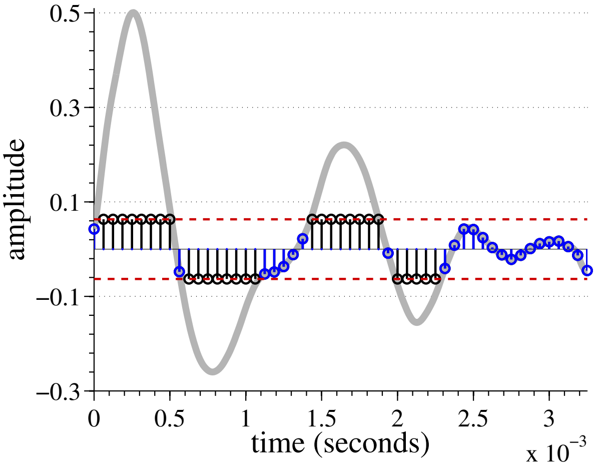


Figure 1: 16 kHz speech signal before and after clipping. The reliable samples after clipping are shown in blue, those that are clipped are shown in black. The original, unclipped signal is shown in grey.

A large number of declipping techniques have been proposed over the years. One of the most common themes is the use of an autoregressive (AR) model to predict the missing samples, e.g., as in linear predictive coding (LPC) [1]. Perhaps the most widely-cited work using AR modeling is by Janssen *et al.* [2], in which LP coefficients are recursively estimated from the clipped speech using an EM-like algorithm. Work by Fong and Godsill utilizes AR modeling, but not directly, instead using AR as the underlying statistical model of a particle filter from which an interpolating sequence of samples is drawn [3]. Selesnick proposed a declipping approach based on minimizing the third derivative of the reconstructed signal [4]. Other more recent approaches include reconstructions based on sparsity ([5], [6]) and recursive vector projection [7]. The work by Kitic *et al.* is particularly successful and motivates some of the techniques in this paper.

The remainder of the paper is organized as follows. Section 2.1 outlines the effect of clipping on automatic speech recognition (ASR) performance. Section 2.2 analyzes what causes

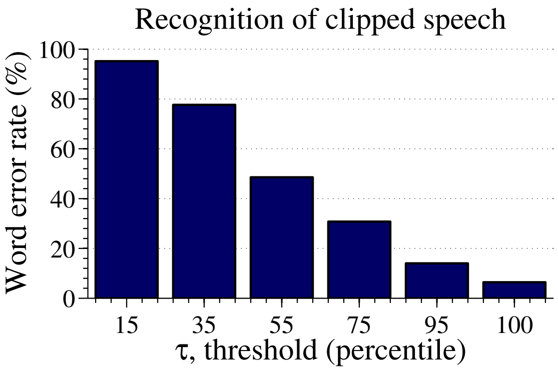


Figure 2: Word error rates for recognition of clipped speech using the CMU Sphinx-III ASR trained on clean, unclipped speech. Note that $\tau = P_{100}$ is no clipping.

classic declipping algorithms to fail. Section 3 introduces Constrained Blind Amplitude Reconstruction, or cBAR, a novel declipping algorithm motivated by the principles outlined in Section 2.2. Finally, Section 4 compares the performance of the cBAR algorithm to current state-of-the-art declipping technology.

2. Motivation

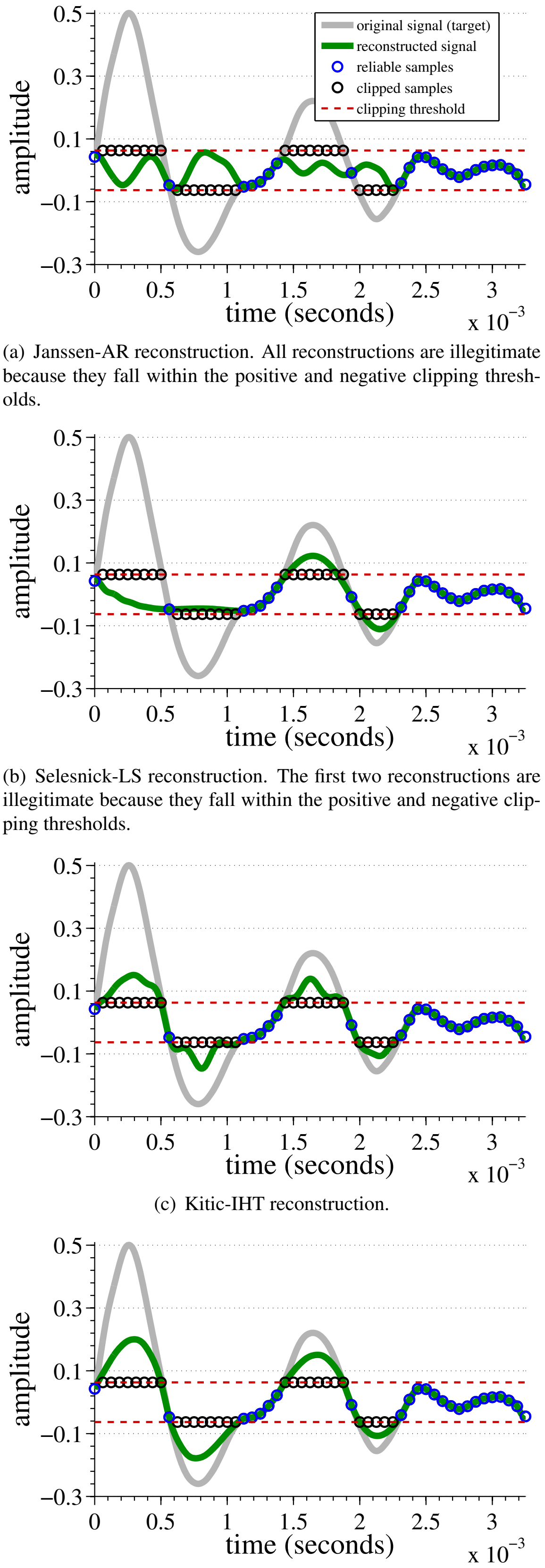
2.1. Effect of clipping on automatic speech recognition

Clipping has an unsurprisingly deleterious effect on ASR performance. Figure 2 shows results of decoding speech that has been clipped at various clipping thresholds[1]. Note that an approximately linear relationship between threshold and word error rate (WER) is observed.

2.2. What makes current declipping algorithms fail?

An effective declipping algorithm that can blindly reconstruct data from a clipped observation could serve to significantly improve the robustness of ASR systems that process clipped speech. Many classic methods fail to accurately reconstruct the clipped signals for all but the most benign clipping thresholds. To illustrate, the following techniques will be considered in more detail:

1. *Janssen-AR*: This classic algorithm proposed by Janssen *et al.* [2] models the clipped observation as an AR process. The per-frame linear prediction coefficients are estimated from the clipped observation using an iterative EM-like algorithm. Given the estimated coefficients, the missing data are interpolated using linear prediction.
2. *Selesnick-LS*: This unpublished technique interpolates the missing data such that the third derivative of the



(a) Janssen-AR reconstruction. All reconstructions are illegitimate because they fall within the positive and negative clipping thresholds.

(b) Selesnick-LS reconstruction. The first two reconstructions are illegitimate because they fall within the positive and negative clipping thresholds.

(c) Kitic-IHT reconstruction.

(d) Constrained BAR reconstruction.

Figure 3: Comparison of current declipping algorithms.

[1] All speech recognition experiments are run using CMU Sphinx-III [8], trained on the clean RM1 database [9], with Mel-Frequency Cepstral Coefficient (MFCC) features. The RM1 database is sampled at 16 kHz and contains 1600 training utterances and 600 test utterances. A standard bigram language model and 8-component GMM-based acoustic model were used. Sphinx-III is an HMM-based system. The MFCC features use a 40-band Mel-spaced triangular filter bank between 133 Hz and 6855 Hz. Windowing of 25.625 ms duration is performed at 100 frames per second using a Hamming window. Utterance-level cepstral mean subtraction is performed before training and decoding.

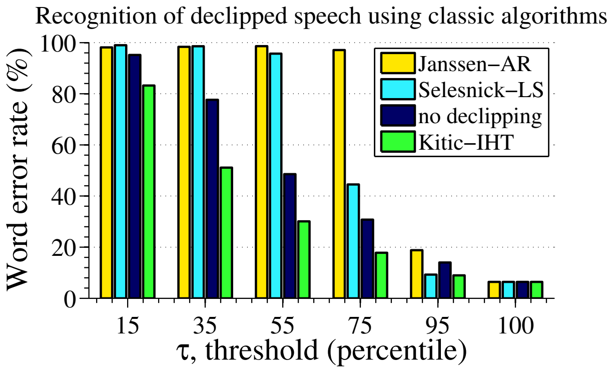


Figure 4: Word error rates for recognition of declipped speech using the CMU Sphinx-III ASR, which had been trained on clean, unclipped speech. The speech was declipped using three standard techniques. *Kitic-IHT* is the only algorithm that is generally effective. Note that $\tau = P_{100}$ is no clipping.

resulting signal is minimized in the least-squares sense [4]. Minimization of the third derivative encourages the interpolated data points to take on a parabolic shape (i.e., because the third derivative of a parabola is zero). A closed-form solution is possible; consequently, this method is typically much faster than the more typical iterative techniques.

3. *Kitic-IHT*: This recently published algorithm is the current state-of-the-art in declipping technology. Continuing with the recent trend toward sparsity-based algorithms, *Kitic-IHT* uses Iterative Hard Thresholding (IHT) to learn a sparse representation of the incoming clipped speech in terms of Gabor basis vectors [6]. The sparse representation is then used to reconstruct the signal on a frame-by-frame basis.

Figures 3a, 3b, and 3c depict the reconstruction of the signal from Figure 1 for each of the three previously-described algorithms, respectively. Similarly, Figure 4 shows WER results of decoding speech after application of the corresponding declipping algorithm. Note that the *Kitic-IHT* algorithm yields an obviously more accurate reconstruction, both graphically and in terms of WER. The difference between *Janssen-AR*, *Selesnick-LS*, and *Kitic-IHT* is that the former two algorithms do not impose any constraints on the reconstructed data, consequently their interpolations are routinely illegitimate, as seen in Figure 3. In general, when declipping, the interpolated data must fall above the clipping threshold, τ, in the absolute sense; moreover, the sign of the interpolated data points should agree with the sign of the observed signal in the same region.

In all results reported in this paper, it is assumed that the value of τ is known *a priori* and that clipped samples can be precisely identified.

3. Constrained blind amplitude reconstruction

The *Constrained Blind Amplitude Reconstruction*, or cBAR, algorithm combines the least squares approach to declipping, motivated by *Selesnick-LS*, with the incorporation of explicit constraints on the values of the interpolated samples, motivated by *Kitic-IHT*. The cBAR algorithm is described as follows.

Define $\boldsymbol{x}$ to be a column vector of length N which contains all the samples of a frame of clipped speech. Suppose there are M reliable samples contained in the vector $\boldsymbol{x}_r$ and $K = N - M$ clipped samples contained in the vector $\boldsymbol{x}_c$. Let $\boldsymbol{S}_r$ be the M x N matrix obtained from the identity matrix by removing all rows corresponding to a clipped sample. Similarly, let $\boldsymbol{S}_c$ be the K x N matrix obtained from the identity matrix by removing all rows corresponding to reliable samples. Finally, let $\boldsymbol{D}_2$ represent the second derivative, a linear operator. Note the following relationship is true [4]:

$$\boldsymbol{x} = \boldsymbol{S}_r^T \boldsymbol{x}_r + \boldsymbol{S}_c^T \boldsymbol{x}_c \tag{2}$$

Declipping can be achieved by solving the following nonlinear constrained optimization problem:

$$\begin{aligned} \underset{\boldsymbol{x}_c}{\text{minimize}} \quad & ||\boldsymbol{D}_2 \left(\boldsymbol{S}_r^T \boldsymbol{x}_r + \boldsymbol{S}_c^T \boldsymbol{x}_c \right) ||_2^2 \\ \text{subject to} \quad & \boldsymbol{x}_c \circ \text{sgn}\, \boldsymbol{S}_c \boldsymbol{x} \geq +\tau \boldsymbol{1} \end{aligned} \tag{3}$$

In the constraint term of Equation 3, the $\circ$ represents the Hadamard (elementwise) product of two vectors or matrices. This particular choice of constraint simultaneously ensures that the interpolated data points will be greater than τ in the absolute sense, and of the correct sign with respect to the observed signal.

The nonlinear optimization problem can be solved using a line search algorithm [10]. A line search is an iterative algorithm that minimizes an objective function, in this case the one defined by Equation 3, by computing a descent direction followed by a step size on each iteration. In our implementation, the descent direction is computed using the quasi-Newton method, the benefit of which is that a full second-order derivative Hessian matrix does not need to be computed. The line search method is an *active-set* method because, on each iteration, the current "active" constraints (i.e., the points which lie on the constraint boundary) are maintained. Knowledge of these points allows one to determine the largest possible step size on each iteration.

We refer to the process of sequentially solving Equation 3 on a frame-by-frame basis as *Constrained Blind Amplitude Reconstruction* or cBAR. In the current implementation, we have set the frame length, N, equal to 80 samples. No frame overlap is used. When the frame boundary falls on a clipped sample point, the length of that particular frame is incremented until the boundary falls on an unclipped sample. Pilot results indicate that a shorter frame length decreases the run time of the algorithm at the expense of a less smooth reconstruction. Further, any derivative (or combination of derivatives) can be minimized in searching for the optimal reconstruction vector. Our preliminary results show that minimizing the second derivative is best; further implications of varying the algorithm's parameters is intended for future research.

4. Results

4.1. Speech recognition performance

In all of the previous literature, the efficacy of declipping algorithms has been measured in terms of mean-squared reconstruction error, or in terms of perceptual listening experiments ([2], [6]). While the results of perceptual listening experiments are very important for pure speech enhancement, the primary aim of this research is to improve the performance of ASR systems. Figure 5 shows comparisons of the WER achieved by recog-

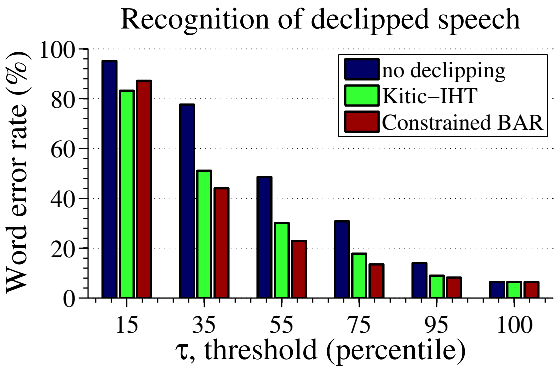


Figure 5: Word error rates for recognition of declipped speech using the CMU Sphinx-III ASR, which had been trained on clean, unclipped speech. This graph indicates that the cBAR algorithm universally improves ASR performance on clipped speech, and further, cBAR outperforms the *Kitic-IHT* method in 4 of the 5 test cases considered. Note that $\tau = P_{100}$ is no clipping.

nizing clipped speech that has been processed by *Kitic-IHT* and cBAR. The cBAR algorithm outperforms *Kitic-IHT* in 4 of the 5 clipping thresholds considered and provides relative WER improvements of up to 56% (at $\tau = P_{75}$) over recognition with no declipping, and up to 24% (also at $\tau = P_{75}$) over recognition with *Kitic-IHT*. A full overview of relative performance improvements is presented in Figure 6. The cBAR algorithm is particularly effective at midrange thresholds of P_{55} and P_{75}.

4.2. Robustness of cBAR

In any practical telecommunications system, some degree of independent additive channel noise is to be expected. This situation is modeled by a system that layers additive white Gaussian noise (AWGN) on top of a clipped speech signal, $x_c[n]$. To evaluate the algorithms' performance in noise, we again assume that the indices of the clipped samples are able to be precisely identified. The noisy estimation of τ and identification of clipped samples in noise is intended for future research. In the experimental results shown in Figure 7, the SNR is measured with respect to the clipped signal, $x_c[n]$, and is set equal to 15 dB. Figure 7 shows that both cBAR and *Kitic-IHT* are generally robust to additive noise, with *Kitic-IHT* slightly more so.

Note that both cBAR and *Kitic-IHT* lower the WER at $\tau = P_{95}$ below its $\tau = P_{100}$ value, and the WER for *Kitic-IHT* at $\tau = P_{75}$ is lower than at $\tau = P_{95}$. This may occur due to the fact that both algorithms tend to generate a smooth reconstruction, which helps to simultaneously reduce the impact of the additive noise.

5. Conclusions

This paper introduced a novel declipping algorithm based on constrained least-squares minimization. It is clear that at a low sampling rate such as 16 kHz, the explicit use of constraints in the declipping process helps to guide a more accurate interpolation. We have shown that the cBAR algorithm outperforms the current state-of-the-art in declipping technology by as much as 24%. The cBAR algorithm is also reasonably robust to additive noise. Future work includes optimization of the parameters that control cBAR (i.e., the window length and the derivative or combination of derivatives used in the objective function), development of a more robust variation of cBAR, and researching methods to accurately estimate clipped samples when additive noise is layered on top of the clipped signal.

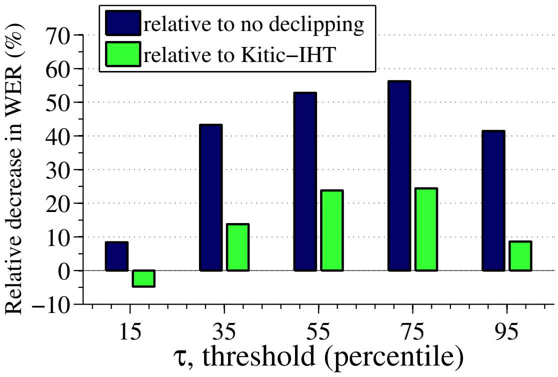


Figure 6: Relative decrease in WER using Constrained BAR. These percentages are derived from the underlying word error rates in Figure 5.

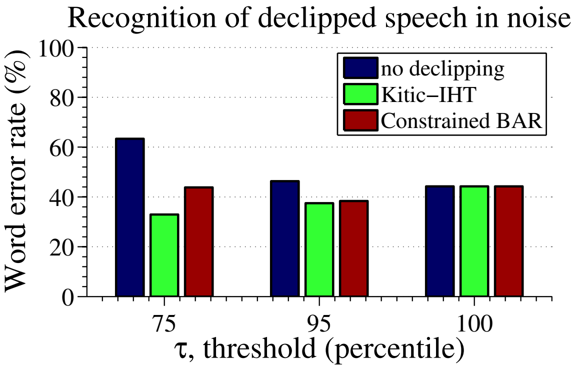


Figure 7: Word error rates for recognition of declipped speech in additive white noise. The SNR with respect to the clipped signal, $x_c[n]$, is 15dB. As before, the ASR had been trained on clean, unclipped speech. Note that $\tau = P_{100}$ is unclipped speech in noise. The smoothness of the reconstructions may help to reduce the additive noise and explain why the WERs for *Kitic-IHT* and cBAR are lower at $\tau = P_{95}$ than at $\tau = P_{100}$.

6. Acknowledgments

This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. D10PC20024. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the view of DARPA or its Contracting Agent, the U.S. Department of the Interior, National Business Center, Acquisition & Property Management Division, Southwest Branch. The authors thank Kornel Laskowski and Ivan Selesnick for many useful discussions, as well as Srdan Kitic for providing implementations of the *Janssen-AR* and *Kitic-IHT* algorithms.

7. References

[1] L. Rabiner and R. Schafer, *Digital Processing of Speech Signals*. Prentice Hall, 1978.

[2] A. Janssen, R. Veldhuis, and L. Vries, "Adaptive interpolation of discrete-time signals that can be modeled as autoregressive processes," *IEEE Trans. on Acoust., Speech and Signal Processing*, pp. 317–330, April 1986.

[3] W. Fong and S. Godsill, "Monte carlo smoothing for nonlinearly distorted signals," in *IEEE Int. Conf. on Acoust., Speech and Signal Processing*, May 2001.

[4] I. Selesnick, "Least squares with examples in signal processing," http://cnx.org/content/m46131/latest/, accessed: 2013-12-11.

[5] A. Adler, V. Emiya, M. Jafari, M. Elad, R. Gribonval, and M. Plumbley, "Audio Inpainting," *IEEE Trans. on Acoust., Speech and Signal Processing*, pp. 922–932, April 2012.

[6] S. Kitic, L. Jacques, N. Madhu, M. Hopwood, A. Spriet, and C. D. Vleeschouwer, "Consistent iterative hard thresholding for signal declipping," in *IEEE Int. Conf. on Acoust., Speech and Signal Processing*, May 2013.

[7] S. Miura, H. Nakajima, S. Miyabe, S. Makino, T. Yamada, and K. Nakadai, "Restoration of clipped audio signal using recursive vector projection," in *TENCON*, November 2011.

[8] P. Placeway, S. Chen, M. Eskenazi, U. Jain, V. Parikh, B. Raj, M. Ravishankar, R. Rosenfeld, K. Seymore, M. Siegler, R. Stern, and E. Thayer, "The 1996 Hub-4 Sphinx-3 System," in *Proceedings of the DARPA Speech Recognition Workshop*, 1997.

[9] P. Price, W. Fisher, J. Bernstein, and D. Pallett, "The DARPA 1000-word resource management database for continuous speech recognition," in *IEEE Int. Conf. on Acoust., Speech and Signal Processing*, April 1988.

[10] *Constrained Nonlinear Optimization Algorithms*, The MathWorks, Natick, MA, 2014. [Online]. Available: http://www.mathworks.com/help/optim/ug/constrained-nonlinear-optimization-algorithms.html