A Modular Approach to Speech Enhancement with an Application to Speech Coding

by

Anthony J. Accardi

Submitted to the Department of Electrical Engineering and Computer Science

in Partial Fulfillment of the Requirements for the Degrees of

Bachelor of Science in Electrical Engineering and Computer Science

and Master of Engineering in Electrical Engineering and Computer Science

at the Massachusetts Institute of Technology

May 26, 1998

Author____
Department of Electrical Engineering and Computer Science
May 17, 1998

Certified by______________________________________
Bernard Gold
Thesis Supervisor

Accepted by______________________________________
Arthur C. Smith
Chairman, Department Committee on Graduate Theses

A Modular Approach to Speech Enhancement with an Application to Speech Coding

by

Anthony J. Accardi

Submitted to the Department of Electrical Engineering and Computer Science

in Partial Fulfillment of the Requirements for the Degrees of

Bachelor of Science in Electrical Engineering and Computer Science

and Master of Engineering in Electrical Engineering and Computer Science

at the Massachusetts Institute of Technology

May 26, 1998

Abstract

We describe two recent single microphone enhancement techniques: Malah's modified MMSE-LSA algorithm and Ephraim's signal subspace approach. We suggest a means of generalizing Malah's design, which makes the introduction of a "core estimator" possible. When we use a modified version of Ephraim's system as such a core estimator, we obtain an enhancement system with improved performance and greater flexibility. We also explore the possibility of using different speech enhancement techniques as pre-processors for different parameter extraction modules of the IS-641 speech coder in order to increase robustness to noise type. We compare these single microphone techniques with a dual microphone alternative that is tailored for an application as a pre-processor for a cell phone. Difficulties encountered with minimum phase acoustic systems suggest that a multi-microphone solution is not necessarily superior to a single microphone enhancement system for arbitrary environmental conditions.

Thesis Supervisor: Bernard Gold
Title: Research Affiliate, MIT

Acknowledgements

This work was done at AT&T Research from 6/97 to 12/97 under the supervision of Richard Cox, as part of the VI-A program at MIT. Rich's guidance and support was invaluable throughout my stay, and he was able to give direction to my efforts whenever I encountered poor results or frustration. I also wish to thank Rich's department at AT&T, where through my interactions with its members, I was able to discover new approaches to my own problems as well as learn a good deal about other ongoing research efforts around me. Jont Allen particularly stands out in this group, as we had many discussions about multi-channel speech enhancement, my experimental setups, the field of hearing research, and countless other signal processing paradigms.

I directly benefited from Yariv Ephraim, who visited AT&T on several occasions. Yariv was the one who originally suggested that I might incorporate his signal subspace approach in my work, and taught me a good deal about spectral estimation and speech enhancement in general. I conversed with David Malah via email throughout the summer, and his aid enabled me to pick up on his modified MMSE-LSA algorithm with the code he left at AT&T in record time.

I'd like to thank my fiancée, Julie, for her criticism of this manuscript. Finally, I'd like to thank my thesis advisor, Bernard Gold, for helping me shape my initial mess of results and speech enhancement techniques into a coherent thesis, and providing helpful criticism of the end product. I am very grateful to all these people for making this thesis a manageable task as well as a great learning experience.

Contents

List of Tables

List of Figures

1

Single Channel Background

- *Spectral Subtraction*
- *Classical Estimation Techniques*
- *Model-Based Approaches*

There are many environments where noisy conditions interfere with speech, such as the inside of a car, a street, or a busy office. The severity of the background noise varies from the gentle hum of a fan inside a computer to cacophonous babble in a crowded café. This background noise not only directly interferes with a listener's ability to understand a speaker's speech, but can cause further unwanted distortions if the speech is encoded or otherwise processed. Speech enhancement is an effort to process the noisy speech for the benefit of the intended listener, be it a human, speech recognition module, or anything else. For a human listener, it is desirable to increase the perceptual quality and intelligibility of the perceived speech, so that the listener understands the communication with minimal effort and fatigue.

It is usually the case that for a given speech enhancement scheme, a tradeoff must be made between the amount of noise removed and the distortion introduced as a side effect. If too much noise is removed, the resulting distortion can result in listeners preferring the original noisy scenario to the enhanced speech. Preferences are based on more than just the energy of the noise and distortion: unnaturally sounding distortions become annoying to humans when just audible, while a certain elevated level of "natural sounding" background noise is well tolerated. Residual background noise also serves to perceptually mask slight distortions, making its removal even more troublesome.

We will broadly define speech enhancement as the removal of additive noise from a corrupted speech signal in an attempt to increase the intelligibility or quality of the speech. We will assume throughout our investigation that the noise and speech are uncorrelated, as is done frequently in the literature.[1] Single channel speech enhancement is the simplest scenario, where only one version of the noisy speech is available, which is typically the result of recording someone speaking in a noisy environment with a single microphone. Multi-channel speech enhancement systems take more than one measurement of the noisy speech and can therefore infer certain properties of the relevant acoustic systems in an attempt to enhance the speech by determining how the noise and speech were mixed. Both paradigms are illustrated in Figure 1.1.

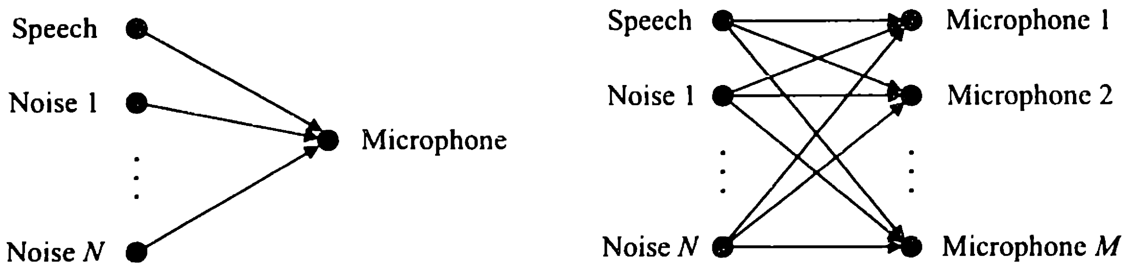


Figure 1.1 Speech enhancement setups for N noise sources: single channel *(left)*, multi-channel with M microphones *(right)*.

For the single channel case, exact reconstruction of the clean speech signal is usually impossible in practice, so speech enhancement algorithms must strike a balance between the amount of noise they attempt to remove and the degree of distortion that is introduced as a side effect. Since any noise component at the microphone cannot in general be distinguished as coming from a specific noise source, we denote the sum of the responses at the microphone from each noise source as a single additive noise term.

Speech enhancement has a number of potential applications. In some cases, a human listener observes the output of the speech enhancement directly, while in others speech enhancement is merely the first stage in a communications channel and might be used as a pre-processor for a speech coder or speech recognition module. Such a variety of different application scenarios places very different demands on the performance of the speech enhancement module, so any speech enhancement scheme must be developed with the intended application in mind. Additionally, many well-known speech enhancement algorithms perform very differently with different speakers and noise conditions, making robustness in design a primary concern. Implementation issues such as delay and computational complexity also arise, although they will not greatly concern us here.

In this chapter, we will survey several classical single channel speech enhancement techniques. Our investigation will be neither comprehensive nor detailed; the purpose is to give the reader a flavor for the myriad techniques applied to solve this problem and a feel for some of the tradeoffs involved.

1.1 Spectral Subtraction

One of the earliest and most widely used methods for speech enhancement is spectral subtraction.[2] This procedure makes use of the fact that the noise characteristics can be measured directly during periods of non-speech activity. Since the noisy speech spectrum can be readily estimated, we can estimate the clean speech spectrum by subtracting the noise spectrum from the noisy speech spectrum.

For an N-length segment of clean speech $x[n]$ that is contaminated by uncorrelated additive noise $w[n]$ to form a noisy speech signal $y[n]$, we have

$$y[n] = x[n] + w[n] \tag{1.1}$$

Taking the Discrete Fourier Transform (DFT) gives

$$Y_k = X_k + W_k \tag{1.2}$$

where

$$x[n] \xleftrightarrow{DFT} X_k$$

$$X_k = \sum_{n=0}^{N-1} x[n] e^{-jn\frac{2\pi k}{N}} \tag{1.3}$$

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X_k e^{jn\frac{2\pi k}{N}} \tag{1.4}$$

We assume that we have an estimate $\hat{W}_k$ of the noise spectrum, and then estimate

$$\hat{X}_k = \left(|Y_k| - |\hat{W}_k|\right) e^{j\angle Y_k} \tag{1.5}$$

This estimator is applied to every frame of the speech. We see from (1.5) that the phase of the noisy speech is kept for the estimate of the clean speech and only the amplitude is changed. Because of these characteristics, spectral subtraction is a member of a class of speech enhancement algorithms that are based on Short-Time Spectral Amplitude (STSA) estimates.[3] There is a theoretical justification for focusing on estimating the amplitude of the clean speech spectrum and retaining the "noisy phase" that we will explore in Chapter 2.

This procedure and all STSA algorithms implicitly place restrictions on the stationarity of the noise, as the noise characteristics must change slowly enough that they can be updated during the infrequent periods of non-speech activity. Another complication is that a Voice Activity Detector (VAD) is required to detect the presence of the speaker's speech in the input signal. This is a non-trivial problem, since aspirations and unvoiced portions of speech can be mistaken for noise, and certain types of noise (e.g. babble) can be mistaken for the primary speaker's speech.

Another serious side effect of spectral subtraction is the production of musical noise in the enhanced output. This problem arises from the time-varying nature of the algorithm. New spectral estimates are obtained for each frame of speech, so the spectral magnitude of the enhanced speech at some given frequency can oscillate wildly about the true value, introducing a tonal distortion that can be more annoying and tiring to the listener than the original noisy speech. This distortion can be improved in a number of ways, such as identifying spectral bands with low speech energy or over-subtracting the noise spectrum using perceptual criteria.[4], [5]

1.2 Classical Estimation Techniques

A number of procedures make use of classical estimation techniques that are applied on a frame-to-frame or adaptive basis and attempt to minimize some distortion measure with respect to the clean speech. Some examples are generalized Wiener filters for estimating clean speech in the time domain in a minimum mean-square error (MMSE) sense[3], the MMSE spectral amplitude estimator[6], and the MMSE log-spectral amplitude (MMSE-LSA) estimator[7].

For each frame of speech, the non-causal Wiener filter for the estimation of the clean speech is calculated to be

$$H(\omega)=\frac{S_{xx}(\omega)}{S_{xx}(\omega)+S_{ww}(\omega)} \tag{1.6}$$

where $S_{xx}(\omega)$ is the power spectral density of $x[n]$.[8] The Wiener filter has zero phase, so the noisy phase is used for the clean speech estimate, just as in spectral subtraction. In order to approximate this non-causal Wiener filter, a VAD is usually used to estimate the power spectrum of the noise, and a variety of techniques can be used to estimate $S_{xx}(\omega)$. Possibilities include subtracting an estimate of $S_{ww}(\omega)$ from a direct estimate of $S_{yy}(\omega)$ (obtained from either averaging or smoothing), or using $|\hat{X}(\omega)|^2$ where $X(\omega)$is estimated separately. The Wiener filter can also be generalized to

$$H(\omega)=\left(\frac{S_{xx}(\omega)}{S_{xx}(\omega)+\alpha\cdot S_{ww}(\omega)}\right)^{\beta} \tag{1.7}$$

where the constants α and β are chosen to obtain different filter characteristics.[3]

Spectral subtraction is based on an optimal variance estimator, while the Wiener filter techniques are derived from the optimal MMSE signal spectral estimator. The MMSE spectral amplitude estimator is the optimal MMSE amplitude estimator.[6] That is, for

$$X_k = A_k e^{j\varphi_k} \tag{1.8}$$

we estimate

$$\hat{A}_k = E[A_k|Y_k] \tag{1.9}$$

The error criterion used when applying such algorithms is essential and attempts to reflect the perceptual quality of the estimate to a human listener. A discussion of several such distortion measures can be found in [9], where measures based on the log-spectral amplitude are determined to be effective. Note that all of these measures merely approximate the human auditory system, and arise due to the tractability of the estimators they result in as well as their perceptual relevance. They will be necessarily less effective for use in speech enhancement than an accurate perceptual hearing model based measure.

1.3 Model-Based Approaches

Another class of speech enhancement algorithms attempts to model the speech in some manner. The physical relevance to speech production varies depending on the model, but generally the results produced by the model are more indicative of the model's performance and usefulness.

The lossless tube model is widely used as a model for speech production, and assumes that the vocal tract can be represented by a concatenation of equal length lossless acoustic tubes, as shown in Figure 1.2. As the number of tubes increases, we can more accurately model a vocal tract with continuously changing area. However this model is not fully physically accurate, since it neglects friction, heat conduction, and wall vibrations. By solving the wave equation, it can be shown that the transfer function of this linear time-invariant model is that of an all-pole system $H(z)$.[10] The shape of the vocal tract slowly varies over time, so in practice we treat $H(z)$ as a time-varying filter. For a p^{th} order model, gain G, and coefficients a_k, we have

$$H(z)=\frac{G}{1-\sum_{k=1}^{p}a_k z^{-k}} \tag{1.10}$$

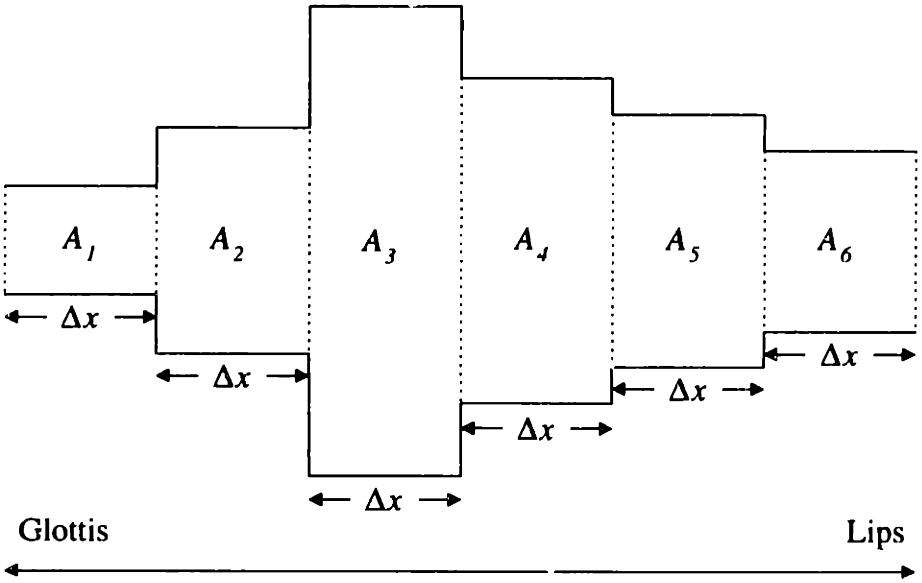


Figure 1.2 Lossless tube model of the vocal tract. Segment k has cross-sectional area A_k and length Δx.

Now we approximate speech production with the model shown in Figure 1.3, where either a train of impulses simulating glottal pulses at some pitch period or random noise is passed through the vocal tract $H(z)$ to produce speech. The result is voiced or unvoiced speech depending on the input type of the filter.

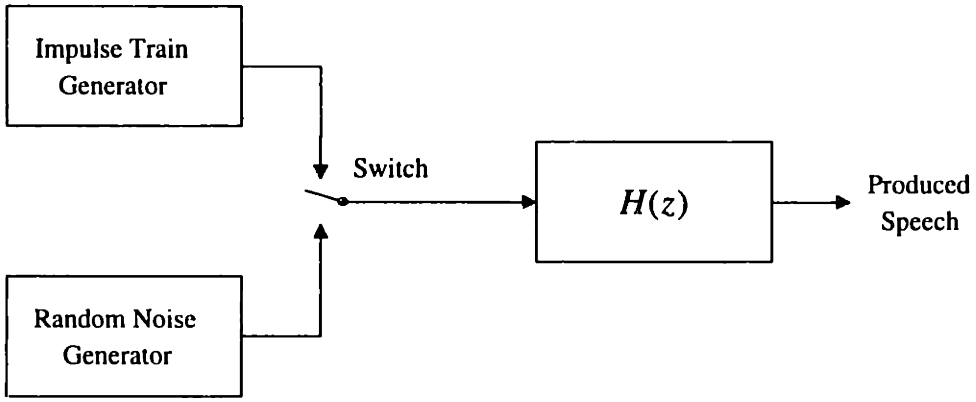


Figure 1.3 A simple discrete-time model of speech production, where $H(z)$ is the transfer function that models the effect of the vocal tract.

This is the basis for Linear Predictive Coding (LPC), and such a model is found in many speech coders.[11] Estimating these LPC parameters a_k based on the maximum a posteriori (MAP) criteria has been studied extensively.[12]

A more useful model in recent literature has been the autoregressive (AR) model. A linear AR model is one where the present value of the signal in consideration can be expressed in terms of a linear combination of its previous values and additive white noise. As an example, for a p^{th} order model of signal $x[n]$ where $e[n]$ is white noise, we have

$$x[n] = \sum_{k=1}^{p} a_k x[n-k] + e[n] \tag{1.11}$$

This is analogous to driving an all-pole system with white noise. The physical interpretation of the AR model as applied to speech is not clear, although a relationship with LPC analysis can seen. The application of AR models to speech has been driven by pragmatic demands, as the resulting speech quality of speech coders and synthesizers based on this model is comparably good.[11]

Scalar and vector Kalman filters have been developed for enhancing speech contaminated with white and colored noise that assume AR models for speech.[13] Such techniques result in improved speech quality and intelligibility both after enhancement and after the enhanced speech has been coded by an LPC-based speech coder.

2

Modified MMSE-LSA Estimator

- *The MMSE-LSA Estimator*
- *Signal Presence Uncertainty*
- *Noise Adaptation*
- *Results*

The modified Minimum Mean-Square Error Log-Spectral Amplitude (modified MMSE-LSA) estimator for speech enhancement is due to David Malah and his work during 1996 at AT&T Research. Malah drew upon three main ideas:

- The Minimum Mean Square Error Log-Spectral Amplitude (MMSE-LSA) estimator.[7]
- The soft decision approach.[14]
- A novel noise adaptation scheme.

The algorithm is a member of the class of STSA enhancement techniques and its structure is shown in Figure 2.1.

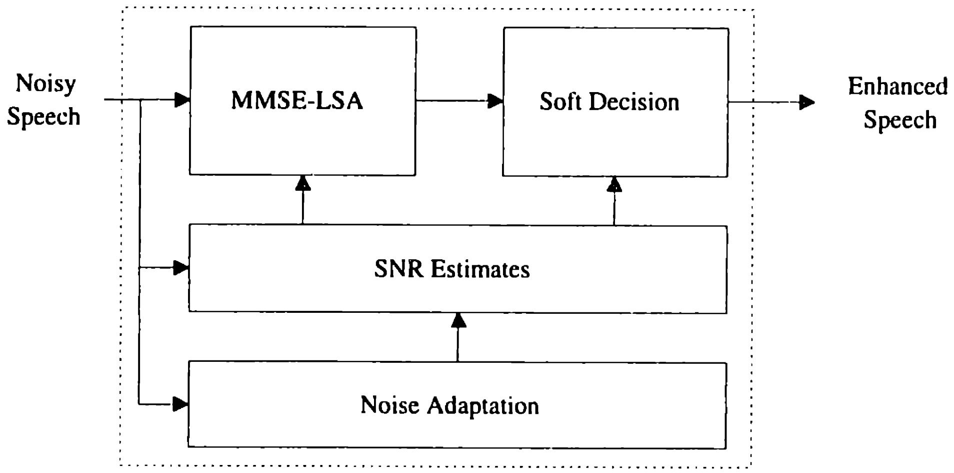


Figure 2.1 Block diagram of the modified MMSE-LSA speech enhancement algorithm.

The MMSE-LSA algorithm operates in the frequency domain and applies a gain to each DFT coefficient of the noisy speech that is computed from signal-to-noise ratio (SNR) estimates. A soft decision module applies an additional gain in the frequency domain that accounts for signal presence uncertainty. A noise adaptation scheme supplies estimates of current noise characteristics for use in the SNR calculations. We will explore each module in depth, and then discuss the effectiveness of the overall system.

2.1 The MMSE-LSA Estimator

We begin by assuming additive independent noise and that the DFT coefficients of both the clean speech and the noise are zero-mean, statistically independent, Gaussian random variables. We formulate the speech enhancement problem as

$$y[n] = x[n] + w[n] \tag{2.1}$$

Taking the DFT of (2.1), we obtain

$$Y_k = X_k + W_k \tag{2.2}$$

We express the complex clean and noisy speech DFT coefficients in exponential form as

$$X_k = A_k e^{j\varphi_k} \tag{2.3}$$

$$Y_k = R_k e^{j\theta_k} \tag{2.4}$$

Now the MMSE-LSA estimate of A_k is the amplitude that minimizes the difference between $\log A_k$ and the logarithm of that amplitude in a MMSE sense:

$$\hat{A}_k = \arg\min_B E[(\log A_k - \log B)^2] \tag{2.5}$$

The solution to (2.5) is the exponential of the conditional expectation[8]:

$$\hat{A}_k = \exp(E[\log A_k | Y_k]) \tag{2.6}$$

Therefore, to implement the MMSE-LSA estimator, we must scale the noisy speech DFT coefficients Y_k so that they have the estimated amplitude $\hat{A}_k$. Our estimate of the clean speech in the frequency domain is now

$$\hat{X}_k = \hat{A}_k \frac{Y_k}{|Y_k|} \tag{2.7}$$

We are using the "noisy phase" in (2.7), as the phase of the DFT coefficients of the noisy speech is used in our estimate of the clean speech. In [6], it is shown that the MMSE complex exponential estimator does not have a modulus of 1. So when an optimal complex exponential estimator is combined with an optimal amplitude estimator, the resulting amplitude estimate is no longer optimal. When the estimate's modulus is constrained to be unity, however, the MMSE complex exponential estimator is the exponent of the noisy phase. In addition, the optimal estimator of the principal value of the phase is the noisy phase itself. This provides justification for using the MMSE-LSA estimator to estimate A_k and to leave the noisy phase untouched, as indicated in (2.7).

The computation of the expectation in (2.6) is non-trivial and presented in [7], where $\hat{A}_k$ is shown to be:

$$\hat{A}_k = G(\xi_k, \gamma_k) \cdot R_k \tag{2.8}$$

where

$$G(\xi_k, \gamma_k) = \frac{\xi_k}{1+\xi_k} \exp\left(\frac{1}{2} \int_{\upsilon_k}^{\infty} \frac{e^{-t}}{t} dt \right) \tag{2.9}$$

$$\upsilon_k = \frac{\xi_k}{1+\xi_k} \gamma_k \tag{2.10}$$

$$\xi_k = \lambda_x(k)/\lambda_w(k) \tag{2.11}$$

$$\gamma_k = R_k^2/\lambda_w(k) \tag{2.12}$$

$$\lambda_x(k) = E[|X_k|^2] = E[|A_k|^2] \tag{2.13}$$

$$\lambda_w(k) = E[|W_k|^2] \tag{2.14}$$

Here $\lambda_x(k)$ and $\lambda_w(k)$ defined in (2.13) and (2.14) are the energy spectral coefficients of the clean speech and the noise, respectively. As defined in (2.11) and (2.12), the quantities ξ_k and γ_k can be interpreted as signal-to-noise ratios. We will denote ξ_k as the a-priori SNR, as it is the ratio of the energy spectrum of speech to that of the noise prior to the contamination of the speech by the noise. Similarly, we will call γ_k the a-posteriori SNR, as it is the ratio of the energy of the current frame of noisy speech to the energy spectrum of the noise, after the speech has been contaminated.

In order to compute $G(\xi_k, \gamma_k)$ as given in (2.9), we must first estimate these SNRs ξ_k and γ_k. As we shall see in Section 2.3, Malah's noise adaptation scheme provides an estimate of $\lambda_w(k)$, so the a-posteriori SNR γ_k is straightforward to estimate since R_k is readily computed from the noisy speech. However, the a-priori SNR ξ_k is somewhat more difficult to estimate. It turns out that the Maximum Likelihood (ML) estimate of ξ_k does not work very well. In [6] the shortcomings of the ML estimate are discussed and a "decision directed" estimation approach is considered. The key idea is that under our assumption of Gaussian DFT coefficients, the a-priori SNR can be expressed in terms of the a-posteriori SNR as

$$\xi_k = E[\gamma_k - 1] \tag{2.15}$$

For each frame of noisy speech, we can then take a convex combination of our two expressions (2.11) and (2.15) for ξ_k, after dropping the expectation operators, to obtain an estimate for the a-priori SNR using previous values of $\hat{A}_k$ and $\hat{\lambda}_k$. For the nth frame we have

$$\hat{\xi}_k(n) = \alpha \frac{\hat{A}_k^2(n-1)}{\hat{\lambda}_w(k,n-1)} + (1-\alpha)P[\hat{\gamma}_k(n) - 1] \tag{2.16}$$

where
$$P[x] = \begin{cases} x & \text{if } x \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

The $P[x]$ function is used to clip the a-posteriori SNR γ_k to 1 if a smaller value is calculated, and $0 \leq \alpha \leq 1$.

This "decision directed" estimate is mainly responsible for the elimination of musical noise artifacts that plague earlier speech enhancement algorithms.[15] The intuition behind this mechanism is that for large a-posteriori SNRs, the a-priori SNR follows the a-posteriori SNR with a single frame delay. This allows the enhancement scheme to adapt quickly to any sudden changes in the noise characteristics that the noise adaptation scheme perceives. However, for small a-posteriori SNRs, the a-priori SNR is a highly smoothed version of the a-posteriori SNR. Since the a-priori SNR has a major impact in determining the gain as seen in (2.9), there are no sudden fluctuations in gain at any fixed frequency from frame to frame when there is a good deal of noise present. This greatly reduces the musical noise phenomenon. Another means of reducing such distortion is described in [16], where perceptual considerations motivate the use of a spectral derivative measure to control smoothing.

We can choose α to trade off between the degree of noise reduction and the overall distortion. In [15] it is noted that α must be close to 1 (> 0.98) in order to achieve the greatest musical noise reduction effect. However, the higher α the more aggressive the algorithm is in removing the residual noise, which causes additional speech distortion. In fact, the easiest way to trade off between aggression and distortion is through changing α, which has the awkward side effect of disturbing the smoothing properties discussed above.

2.2 Signal Presence Uncertainty

The above analysis assumes that there is speech present in every frequency bin of every frame of the noisy speech. This is generally not the case, and there are two well-established ways of taking advantage of this situation.

The first, called "hard decision", treats the presence of speech in some frequency bin as a time-varying deterministic condition that can be determined using classical detection theory. The second, "soft decision", treats the presence of speech as a stochastic process with a changing binary probability distribution.[14] The soft decision approach has been found to be more successful in speech enhancement.[6] A hard decision approach can in fact lead to musical noise. When the decision oscillates between signal presence and absence in time for some frequency bin, an enhancement scheme that greedily eliminates frequency components containing only noise would produce tonal artifacts at that frequency. Following this outline, we define two states for each frequency bin k. H_0^k denotes the state where the speech signal is absent in the kth bin, while H_1^k is the state where the signal is present in the kth bin. Now our estimate of $\log A_k$ is given by

$$E[\log A_k | Y_k, H_1^k] \Pr(H_1^k | Y_k) + E[\log A_k | Y_k, H_0^k] \Pr(H_0^k | Y_k) \qquad (2.17)$$

Since $E[\log A_k | Y_k, H_0^k] = 0$, soft decision entails weighting our previous estimate of $\log A_k$ by $\Pr(H_1^k | Y_k)$. To compute this weighting factor, we first expand $\Pr(H_1^k, Y_k)$ in two different ways:

$$\Pr(H_1^k | Y_k) \cdot \Pr(Y_k) = \Pr(Y_k | H_1^k) \cdot \Pr(H_1^k) \tag{2.18}$$

Also,

$$\Pr(Y_k) = \Pr(Y_k | H_1^k) \cdot \Pr(H_1^k) + \Pr(Y_k | H_0^k) \cdot \Pr(H_0^k) \tag{2.19}$$

From (2.18) and (2.19) we can solve for $\Pr(H_1^k | Y_k)$ and express it in terms of the likelihood function $\Lambda(k)$:

$$\Pr(H_1^k | Y_k) = \frac{\Lambda(k)}{1 + \Lambda(k)} \tag{2.20}$$

where

$$\Lambda(k) = \mu_k \frac{\Pr(Y_k | H_1^k)}{\Pr(Y_k | H_0^k)} \tag{2.21}$$

$$\mu_k = \Pr(H_1^k) / \Pr(H_0^k) = \frac{1 - q_k}{q_k} \tag{2.22}$$

Here q_k is the a-priori probability of signal absence in the k^{th} bin, and $\Lambda(k)$ is clearly the likelihood function from classical detection theory.[8] With our Gaussian distribution assumptions on Y_k, it is straightforward to calculate $\Lambda(k)$:

$$\Lambda(k) = \frac{1 - q_k}{q_k} \cdot \frac{1}{1 + \eta_k} \exp\left(\frac{\eta_k}{1 + \eta_k} \gamma_k \right)_{\eta_k = \frac{\xi_k}{1 - q_k}} \tag{2.23}$$

where the SNRs γ_k and ξ_k can be estimated in the same manner as described in Section 2.1.

2.3 Noise Adaptation

An important development for the modified MMSE-LSA technique is the noise adaptation scheme, which allows the speech enhancement algorithm to handle non-stationary noise. The adaptation proceeds in two steps; the first identifies all the spectral coefficients in the current frame that are reasonably good representations of the noise, and the second adapts the current noise estimate to this new information.

Direct spectral information about the noise can become available when a frame of the noisy speech is a "noise-only" frame, meaning that the speech contribution during that time period is negligible. In this case, the entire noise spectrum estimate can be updated. Additionally, even if a frame contains both speech and noise, there may still be some "noise-only" frequency bins so that the speech contribution within certain frequency ranges is negligible during the current frame. Here we can update the corresponding spectral components of our noise estimate accurately.

The process of deciding whether a given frame is a noise-only frame is dubbed "segmentation", and the decision is based on the a-posteriori SNR estimates γ_k. Under our Gaussian distribution assumptions on Y_k, we can compute the probability density function $f(\gamma_k)$ for γ_k, which turns out to be an exponential distribution with mean and standard deviation $1+\xi_k$ given by

$$f(\gamma_k)=\frac{1}{1+\xi_k}\exp\left(\frac{-\gamma_k}{1+\xi_k}\right) \tag{2.24}$$

We declare a frame of speech to be noise-only if both our average (over k) estimate of the a-posteriori SNRs is low and the average of our estimate of the variance of the a-posteriori SNR estimator is low. That is, a frame is noise-only when

$$\bar{\gamma}\le\bar{\gamma}_{Threshold} \quad \text{and} \quad \bar{\xi}\le\sigma_{Threshold}-1 \tag{2.25}$$

When a noise-only frame is discovered, we update all the spectral components of our noise estimate by averaging our estimates for the previous frame with our new estimates. So our noise spectral estimate for the k^{th} frequency bin and the n^{th} frame is given by:

$$\hat{\lambda}_w(k,n)=\alpha_w\hat{\lambda}_w(k,n-1)+(1-\alpha_w)R_w^2 \tag{2.26}$$

where α_w is the forgetting factor of the update equation, which is dynamically updated based on the average estimate of γ_k. In this manner, the forgetting factor is directly related to the current value of $\bar{\gamma}$ so that the lower $\bar{\gamma}$ is, the better our estimate of the noise spectrum, and therefore we discard our previous noise spectral estimates more quickly.

The situation for dealing with noise-only frequency bins for frames with signal present is quite similar, except the individual SNR estimates for each frequency bin are used instead of their averages. There is one main difference; since we have an estimate of the probability that each bin contains no signal present (q_k from our soft decision discussion in Section 2.2), we can use this to refine our update of the forgetting factor for each frequency bin.

The impact of this noise adaptation scheme is dramatic. The complete modified MMSE-LSA enhancement algorithm is capable of adapting to great changes in noise volume in only a few frames of speech, and has demonstrated promising performance in dealing with highly non-stationary noise, such as music. Figure 2.2 illustrates the performance of the noise adaptation module on a sentence pair contaminated by 20 dB music (music that is 20 dB below the level of the speech).

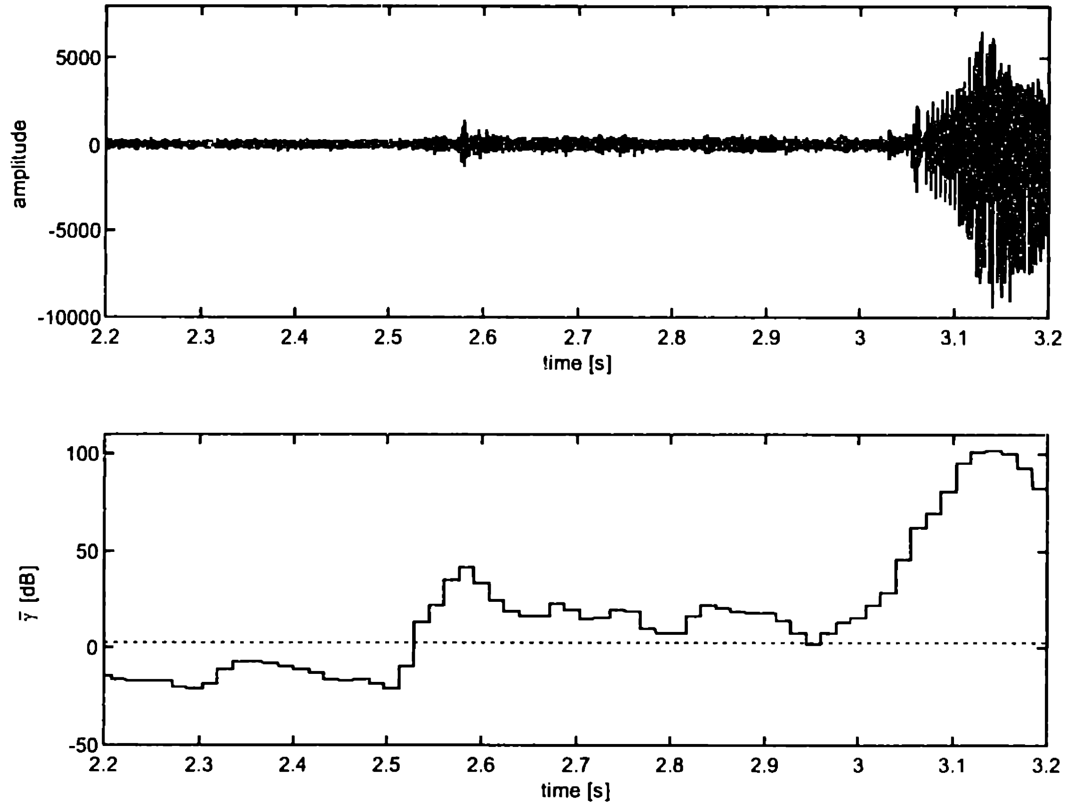


Figure 2.2 A second of noisy speech (20 dB music) in between two sentences in a sentence pair. The beginning of the second sentence is around 3.05 seconds and starts with the word "The" *(above)*. The average a-posteriori SNR $\bar{\gamma}$ with a constant $\bar{\gamma}_{Threshold} = 2\ln 2$ included for reference *(below)*.

We see that the noise adaptation scheme begins to correctly identify the noise-only segments at some point after the end of the first sentence, but between 2.5 and 2.6 seconds into the noisy speech, a new instrument is introduced in the background music noise that throws the adaptation scheme off. The scheme gradually adapts to the new noise characteristics, but does not revert to noise-only segments before the next sentence begins.

2.4 Results

Mean Opinion Score (MOS) tests were conducted for the modified MMSE-LSA speech enhancement algorithm. These tests measure the subjective quality of speech by having a panel of listeners rate a speech sample on a 1-5 scale, from poorest quality to best. The average score over all listeners is denoted the "MOS score" for the speech sample. For our tests, six speakers, three male and three female, produced the clean speech. This clean speech was then contaminated with a number of noise sources at different intensity levels. Some enhanced speech samples were then coded with the IS-641 speech coder, which is a 7.4 kbps coder and is described later in Chapter 6. The enhancement algorithm's aggression level was tuned through α for two different settings. The R16 setting was optimized for use as a pre-processor for a 16 kbps speech coder (corresponding to $\alpha = 0.91$) and the R8 setting, a more aggressive one, was tuned as a pre-processor for an 8 kbps speech coder ($\alpha = 0.95$).

Relevant MOS scores are given below in Table 2.1. On the average over all pairs of conditions, MOS scores that differ by more than 0.14 are considered statistically different with a 95% level of confidence. This number can be computed using analysis of variance (ANOVA) techniques.[17]

	Clean	Babble 20dB	Car 10dB	Car 15dB	Car 20dB
None	4.155	3.702	2.915	3.442	3.694
R16	4.256	3.864	3.089	3.593	3.837
None, IS-641	3.899	3.581	2.806	3.132	3.438
R8, IS-641	3.923	3.419	2.663	3.174	3.485

Table 2.1 MOS scores for different enhancement types (by row) and different noise types and intensities (by column). Those enhanced speech samples that were subsequently coded by IS-641 are denoted.

The results are quite disappointing. The noisy, unenhanced, coded output was either preferred over the enhanced version at the R8 setting or the two were determined to be of similar quality for all of the noise types and levels. The distortion introduced by the enhancement was particularly devastating for high noise intensities and for babble. However, MOS test data for other coders (not shown) demonstrate that the R16 setting is preferred over the noisy coded version, and we see in Table 2.1 that enhancement with the R16 setting is always preferred over the noisy signal when no speech coder is involved. This suggests that R8 is too aggressive an enhancement scheme, and that the test subjects preferred the extra residual noise in the unenhanced speech over the distortions introduced by the enhancement algorithm. MOS test data is not available for the R16 setting and the IS-641 coder for this test session, but we will show that our hypothesis here is correct in Chapter 5 with the results of a more recent MOS test.

3

A Signal Subspace Approach

- *Time Domain Constrained Estimator*
- *Spectral Domain Constrained Estimator*
- *Reverse Spectral Domain Constrained Estimator*
- *Results*

Yariv Ephraim and Harry L. Van Trees developed a signal subspace approach[18] that provides a theoretical framework for understanding a number of classical speech enhancement techniques, and allows for the application of external criteria to control enhancement performance. The basic idea is that the vector space of the noisy speech can be decomposed into a signal-plus-noise subspace and a noise-only subspace. Once identified, the noise-only subspace can be eliminated and then the speech estimated from the remaining signal-plus-noise subspace. We assume that the full space has dimension K and the signal-plus-noise subspace has dimension $M < K$.

Say we have clean speech $x[n]$ that is corrupted by independent additive noise $w[n]$ to produce a noisy speech signal $y[n]$. We constrain ourselves to estimating $x[n]$ using a linear filter $\mathbf{H}$, and will initially consider $w[n]$ to be a white noise process with variance σ_w^2. In vector notation, we have

$$\mathbf{y} = \mathbf{x} + \mathbf{w} \tag{3.1}$$

$$\hat{\mathbf{x}} = \mathbf{H}\mathbf{y} \tag{3.2}$$

We can decompose the residual error into a term solely dependent on the clean speech, called the signal distortion $\mathbf{r}_x$, and a term solely dependent on the noise, called the residual noise $\mathbf{r}_w$:

$$\begin{aligned} \mathbf{r} &= \hat{\mathbf{x}} - \mathbf{x} \\ &= (\mathbf{H} - \mathbf{I})\mathbf{x} + \mathbf{H}\mathbf{w} \\ &= \mathbf{r}_x + \mathbf{r}_w \end{aligned} \tag{3.3}$$

In (3.3) we have explicitly identified the tradeoff between residual noise and speech distortion. Since different applications could require different tradeoffs between these two factors, it is desirable to perform a constrained minimization using functions of the distortion and residual noise vectors. Then the constraints can be selected to meet the application requirements.

3.1 Time Domain Constrained Estimator

Two different frameworks for performing a constrained minimization using functions of the residual noise and signal distortion are presented in [18]. The first examines the energy in these vectors and results in a time domain constrained estimator. We define

$$\bar{\varepsilon}_x^2 = \operatorname{tr} E[\mathbf{r}_x \mathbf{r}_x^{\#}] = \operatorname{tr}\{(\mathbf{H} - \mathbf{I})\mathbf{R}_y(\mathbf{H} - \mathbf{I})^{\#}\} \tag{3.4}$$

to be the energy of the signal distortion vector $\mathbf{r}_x$, and similarly define

$$\bar{\varepsilon}_w^2 = \operatorname{tr} E[\mathbf{r}_w \mathbf{r}_w^{\#}] = \sigma_w^2 \operatorname{tr}\{\mathbf{H}\mathbf{H}^{\#}\} \tag{3.5}$$

to be the energy of the residual noise vector $\mathbf{r}_w$.

We desire to minimize the energy of the signal distortion while constraining the energy of the residual noise to be less than some fraction $K\alpha$ of the noise variance $\sigma_{\mathbf{w}}^2$:

$$\min_{\mathbf{H}} \bar{\varepsilon}_{\mathbf{x}}^2 \quad \text{subject to} \quad \bar{\varepsilon}_{\mathbf{w}}^2 / K \leq \alpha\sigma_{\mathbf{w}}^2 \tag{3.6}$$

The solution to the constrained minimization problem in (3.6) involves first the projection of the noisy speech signal onto the signal-plus-noise subspace, followed by a gain applied to each eigenvalue, and finally the reconstruction of the signal from the signal-plus-noise subspace. The gain for the m^{th} eigenvalue is a function of the Lagrange multiplier μ, and is given by

$$g_\mu(m) = \frac{\lambda_{\mathbf{x}}(m)}{\lambda_{\mathbf{x}}(m) + \mu\sigma_{\mathbf{w}}^2} \tag{3.7}$$

where $\lambda_{\mathbf{x}}(m)$ is the m^{th} eigenvalue of the clean speech.

Thus, the enhancement system can be interpreted as a Karhunen-Loève Transform (KLT), followed by a set of gains, and ending with an inverse KLT. This system is shown in Figure 3.1.

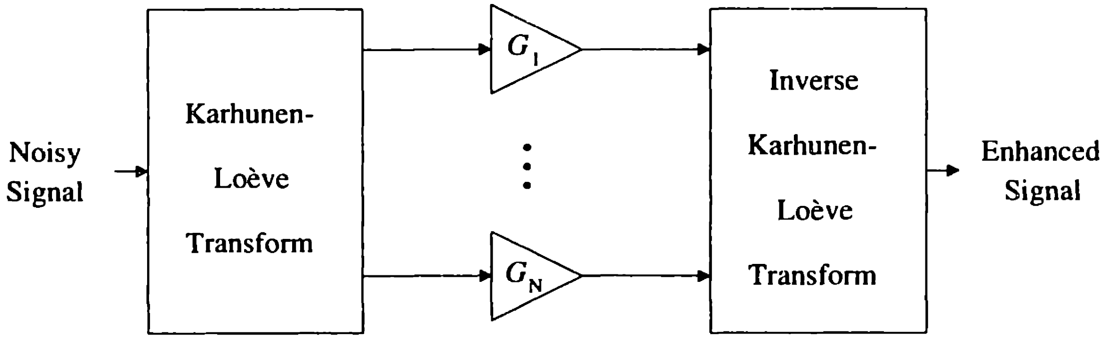


Figure 3.1 Block diagram of an implementation of a signal subspace estimator.

Ephraim shows that μ is uniquely determined by our choice of the constraint α, and demonstrates how the generalized Wiener filter in (3.7) can implement linear MMSE estimation and spectral subtraction for specific values of μ and certain approximations to the KLT.

3.2 Spectral Domain Constrained Estimator

Motivated by this framework, it would seem desirable to provide a tighter means of control over the tradeoff between residual noise and signal distortion. Toward this end, Ephraim derives a spectral domain constrained estimator[18] which minimizes the energy of the signal distortion while constraining each of the eigenvalues of the residual noise by a different constant proportion of the noise variance:

$$\min_{\mathbf{H}} \bar{\varepsilon}_{\mathbf{x}}^2 \quad \text{subject to} \quad E[|\mathbf{u}_k^{\#}\mathbf{r}_{\mathbf{w}}|^2] \le \alpha_k \sigma_{\mathbf{w}}^2 \tag{3.8}$$

Here $\mathbf{u}_k$ is the k^{th} eigenvector of the noisy speech, $0 \le \alpha_k \le 1$, and the constraint is applied for each k in the signal-plus-noise subspace. The form of the solution to this constrained minimization is very similar to the time domain constrained estimator illustrated in Figure 3.1; the only difference is that the eigenvalue gains are given by

$$g(m) = \sqrt{\alpha_k} \tag{3.9}$$

instead of the result in (3.7).

Now with such freedom over the constraints α_k, the difficulty arises as to how to optimally choose these constants to obtain a reasonable speech enhancement system. One choice Ephraim investigated is

$$\alpha_k = \exp\{-\upsilon\sigma_{\mathbf{w}}^2 / \lambda_{\mathbf{x}}(k)\} \tag{3.10}$$

where υ is a constant that determines the level of noise suppression, or the aggression level of the enhancement algorithm. The constraints in (3.10) effectively shape the noise so it resembles the clean speech, which takes advantage of the masking properties of the human auditory system. We will discuss more accurate perceptually based measures in Chapter 4. Another important point is that this choice of functional form for α_k is an aggressive one.

There is no treatment of noise distortion in this signal subspace approach, and it turns out that the residual noise in the enhanced signal can contain artifacts so annoying that the result is less desirable than the original noisy speech. Therefore, when using this signal subspace framework, it is desirable to aggressively reduce the residual noise at the possibly severe cost of increased signal distortion.

3.3 Reverse Spectral Domain Constrained Estimator

As we will discover in Chapter 5, the spectral domain constrained estimator can be placed in a framework that will substantially reduce the noise distortion. In such scenarios, it might be advantageous to use a variant of Ephraim's spectral domain constrained estimator. We will derive this new estimator here. Unlike the spectral domain constrained estimator, we minimize the residual noise with the signal distortion constrained:

$$\min_{\mathbf{H}} \bar{\varepsilon}_{\mathbf{w}}^2 \quad \text{such that} \quad E[|\mathbf{u}_k^{\#}\mathbf{r}_{\mathbf{y}}|^2] \le \alpha_k \lambda_{\mathbf{y},k} \tag{3.11}$$

Since $\mathbf{H}$ could have complex entries, we set the Jacobians of both the real and imaginary parts of the Lagrangian from (3.11) to zero in order to obtain the first order conditions, expressed in matrix form as

$$\mathbf{H}\mathbf{R}_{\mathbf{w}} + \mathbf{U}\Lambda_{\mu}\mathbf{U}^{\#}(\mathbf{H}-\mathbf{I})\mathbf{R}_{\mathbf{y}} = \mathbf{0} \tag{3.12}$$

where $\Lambda_{\mu} = \text{diag}(\mu_1,\dots,\mu_K)$ is a diagonal matrix of Lagrange multipliers and the columns of $\mathbf{U}$ contain the eigenvectors $\mathbf{u}_k$. Applying the eigendecomposition of $\mathbf{R}_{\mathbf{y}}$ and using the assumption that the noise is white, we obtain:

$$\sigma_{\mathbf{w}}^2\mathbf{Q} + \Lambda_{\mu}\mathbf{Q}\Lambda_{\mathbf{y}} = \Lambda_{\mu}\Lambda_{\mathbf{y}} \tag{3.13}$$

where

$$\begin{aligned}&\mathbf{Q} = \mathbf{U}^{\#}\mathbf{H}\mathbf{U}\\ &\Lambda_{\mathbf{y}} = \text{diag}(\lambda_{\mathbf{y},1},\dots,\lambda_{\mathbf{y},K})\end{aligned} \tag{3.14}$$

We note that a possible solution to the constrained minimization is obtained when $\mathbf{Q}$ is diagonal with elements given by

$$q_{kk} = \begin{cases} \dfrac{\mu_k\lambda_{\mathbf{y},k}}{\sigma_{\mathbf{w}}^2 + \mu_k\lambda_{\mathbf{y},k}} & k = 1,\dots,M \\ 0 & k = M+1,\dots,K \end{cases} \tag{3.15}$$

which satisfies (3.13). For this $\mathbf{Q}$, we have

$$E[|\mathbf{u}_k^{\#}\mathbf{r}_{\mathbf{y}}|^2] = \lambda_{\mathbf{y},k}(q_{kk}-1)^2 \tag{3.16}$$

Now for the non-zero constraints in (3.11) to hold with equality, we must have

$$q_{kk} = 1 - \sqrt{\alpha_k} \tag{3.17}$$

and

$$\mu_k = \frac{\sigma_{\mathbf{w}}^2}{\lambda_{\mathbf{y},k}\sqrt{\alpha_k}}(1-\sqrt{\alpha_k}) \tag{3.18}$$

Since we see from (3.18) that $\mu_k \geq 0$, this proposed solution satisfies the Kuhn-Tucker necessary conditions for the constrained minimization.

We conclude that $\mathbf{H}$ is given by

$$\begin{aligned}&\mathbf{H} = \mathbf{U}\mathbf{Q}\mathbf{U}^{\#}\\ &\mathbf{Q} = \text{diag}(q_{11},\dots,q_{KK})\\ &q_{kk} = \begin{cases} 1-\sqrt{\alpha_k} & k = 1,\dots,M \\ 0 & k = M+1,\dots,K \end{cases}\end{aligned} \tag{3.19}$$

Thus the reverse spectral domain constrained estimator has a form very similar to that of our previous signal subspace estimators. The implementation of (3.19) is given in Figure 3.1 with the gains

$$g(m) = 1 - \sqrt{\alpha_k} \tag{3.20}$$

3.4 Results

Ephraim and Van Trees conducted subjective listening tests with 16 listeners and 12 sentences that were contaminated with 10 dB white Gaussian noise.[18] There were two listening sessions; the first compared the signal subspace approach with the plain noisy speech, and the second compared the signal subspace approach with spectral subtraction. The listeners were asked to choose which sentence out of a pair they preferred. The spectral domain constrained estimator was used with an aggression level of $\upsilon = 5$. An initial segment of noise was used to obtain a noise estimate, and the noise was whitened prior to being processed by the algorithm.

Ephraim and Van Trees determined that subjects voted in favor of the signal subspace approach over the noisy signal for 83.9% of the sentences and concluded that the quality of the enhanced signal is far better than the original noisy speech. They reasoned that this was because of the great reduction in noise level despite the distortion introduced by the enhancement algorithm. The voting was also quite dependent on the listener; the two subjects who preferred the noisy signal did so for 66.7%of the sentences on average. For these subjects, the noisy signal sounded more natural and therefore of better quality.[18]

When compared with spectral subtraction, Ephraim and Van Trees reported that the signal subspace approach was preferred for 98.2% of the sentences. The musical noise in the spectral subtraction algorithm was the major cause for this result, as the distortion in the signal subspace method is less annoying.[18]

4

Perceptual Hearing

- *A Masking Model*
- *Application to the Modified MMSE-LSA Algorithm*
- *Application to the Signal Subspace Algorithm*

When optimizing a speech enhancement algorithm where a human listener will directly observe the output, it is important to consider the perceptual quality of the signal. Toward this end, we can make use of various perceptual measures such as loudness and masking. Loudness is a scalar number and is defined as the magnitude of an auditory sensation, and the masking threshold induced by some auditory stimulus is a function of frequency that describes the sound pressure level (SPL) below which no additional stimulus can be heard.[19] The stimulus inducing the masking threshold is denoted as the "masker". In order to make use of such measures, we need a practical method of calculating them. The complete underlying phvsical and psychophysical mechanisms that are responsible for these measures are presently unknown. In modern applications, it is common to approximate these effects by developing efficient algorithms that attempt to match human listening data.[20], [21], [22]

4.1 A Masking Model

A somewhat accurate estimate of the masking threshold can be obtained as Terhardt did in [23] and [24]. The masking threshold induced by a sine wave at various frequencies and sound pressure levels is well known through empirical observations [19]. In [23], the masking threshold for a signal is approximated by considering each Fourier component as a pure tone, calculating the masking threshold induced by each tone, and summing the thresholds. The result must then be normalized so the masking threshold is invariant to the DFT size chosen.

The audible frequency range is divided into a number of different intervals known as critical bands. Through experimentation, it was discovered that the loudness of stimuli in different critical bands adds while for stimuli in the same critical band, the acoustic intensities add instead. In this manner the auditory system acts as a type of nonlinear filter bank with bandwidths determined by these critical bands. The Bark scale is an absolute frequency scale that indicates the edges of the critical bands.[25] An approximate conversion between z Barks and f hertz is

$$z = 13.3\arctan(7.5\times 10^{-4} f) \qquad (4.1)$$

In [23], Terhardt approximates the masking threshold due to a tone as dropping off linearly on a Bark scale with a slope of

$$S_1 = 27 \quad \text{dB / Bark} \qquad (4.2)$$

for frequencies less than the masker frequency. For frequencies greater than the masker frequency, the masking threshold exhibits a more complicated behavior that depends on both the masker frequency f_m and the sound pressure level of the masker L_m in dB. The masking threshold is still approximately linear but with an absolute slope of

$$S_2 = 24 + 230/f_m - 0.2L_m \quad \text{dB / Bark} \tag{4.3}$$

Thus we have a complete masking threshold model for pure tone maskers. Figure 4.1 illustrates two tones, each with a sound pressure level of 80 dB, and the masking threshold calculated for each tone is given by the solid triangular-shaped waveforms.

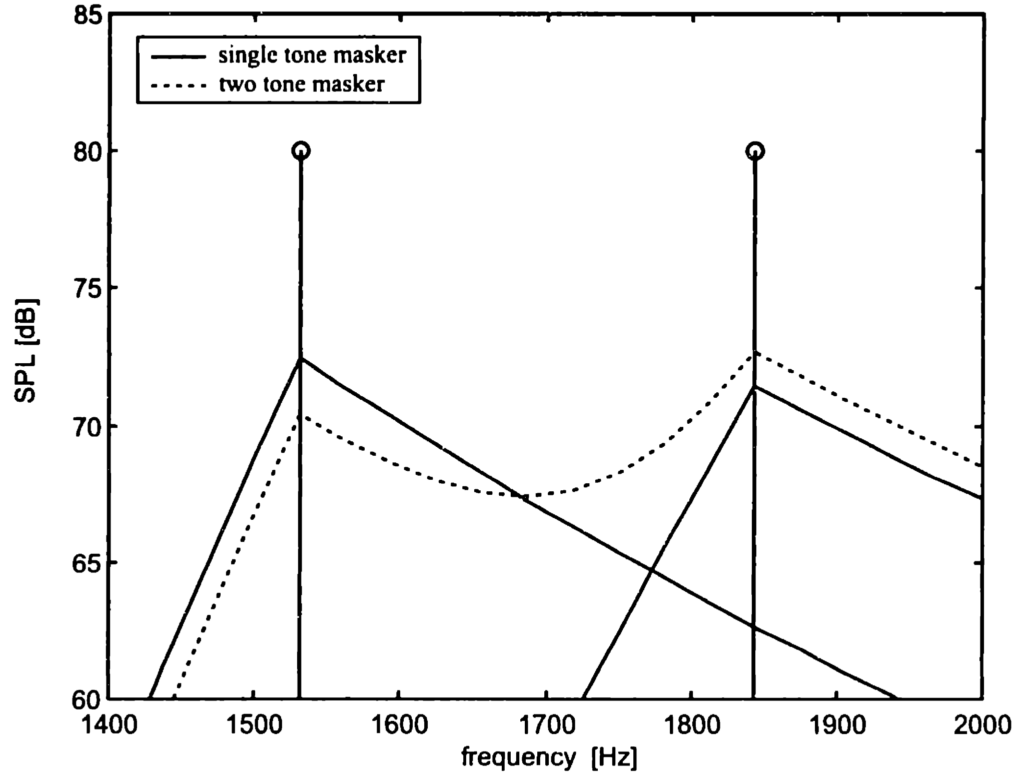


Figure 4.1 Two tone maskers at 80 dB SPL. The masking thresholds computed for each tone separately and for the two tones together are shown.

Terhardt's idea is to extend this computation to handle several tonal maskers by summing the amplitudes of the masking thresholds for each tone. He notes that this is an ad-hoc solution, as the empirically measured masking threshold for a number of stimuli is more complicated and depends on the relative phases of the individual stimuli.[23] In order to compute a masking threshold for an arbitrary signal, we have applied this technique by taking a DFT of the signal and adding the amplitudes of the masking thresholds for each DFT bin, treating each bin as a tonal masker. The obvious difficulty with this procedure is that by taking larger and larger DFTs, the masking threshold can be made arbitrarily high. To address this problem we introduce a subsequent normalization step, where the masking threshold is uniformly scaled so that its energy is equal to the energy of the input sound pressure signal. The complete masking threshold calculated in this manner for two tones is given in Figure 4.1, and for a frame of speech in Figure 4.2. Note in Figure 4.1 how the two-tone masking threshold is greater near the higher frequency tone. This is a consequence of the unequal slopes of the individual masking thresholds given in (4.3).

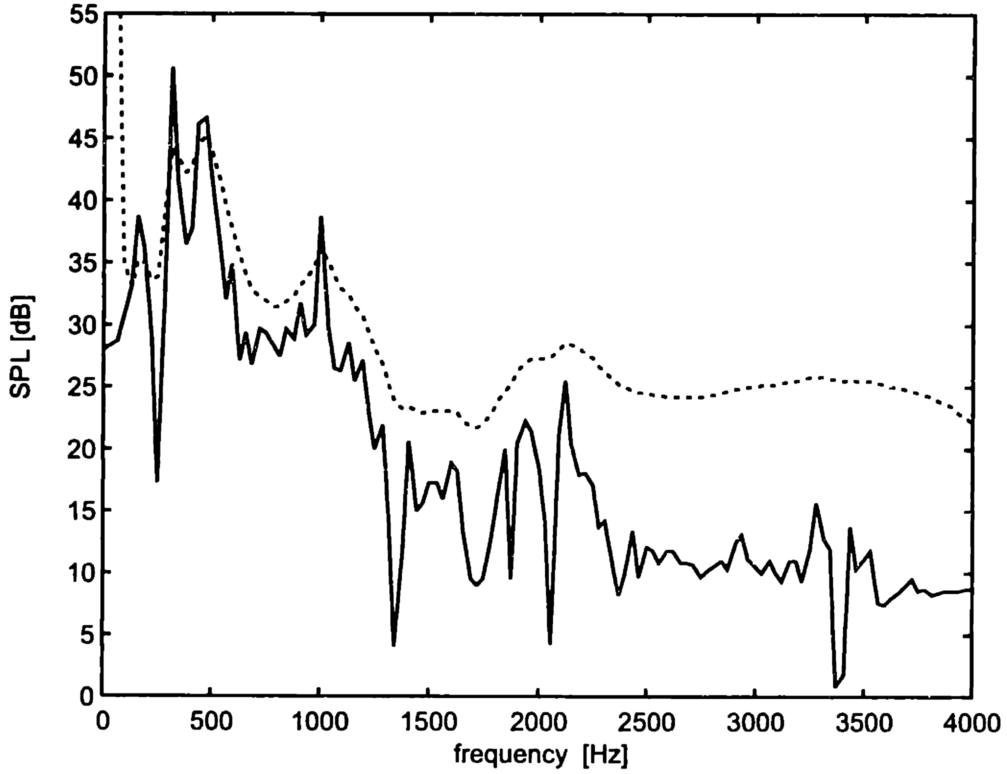


Figure 4.2 The magnitude of the DFT of a frame of speech *(solid)* with the associated computed masking threshold *(dashed)*.

4.2 Application to the Modified MMSE-LSA Algorithm

Signal distortion in the modified MMSE-LSA speech enhancement algorithm is caused by the over-attenuation of certain frequency bands that contain a substantial clean speech component. This prevents us from achieving a large degree of noise suppression in practice since the introduced distortion becomes annoying and degrades the quality of the enhanced signal. We will attempt to use our masking model to identify situations where the aggressive attenuation of a frequency band will only lead to signal distortion and should therefore be prevented.

Toward this end, we use the masking model discussed in Section 4.1 to detect whether the signal and noise in each frequency band is masked by the clean speech alone or remains audible. Ideally our masker would be the output of the enhancement algorithm, but for computational complexity reasons we would prefer an open-loop system. Therefore we approximate this masker by first running the modified MMSE-LSA algorithm on the noisy speech input to obtain an estimate of the masker, and then use this estimate for our masking threshold calculations. In this manner we have four masking categories for each frequency bin, as shown in Table 4.1.

If the noise is masked, no noise suppression is necessary. Any attenuation of the corresponding frequency band will result in signal distortion without any noticeable reduction in noise, so it would be more effective to leave the band alone. However, when the noise is audible in some frequency band, we must consider whether the signal is audible also. If the signal is masked, attenuating the band has no effect on signal distortion. One might think that removing the band entirely would be the best course of action, but such a "perceptual hard decision" (recall our discussion in Section 2.2) would introduce noise structuring and distortion. So in this case we use the old gain computed by the modified MMSE-LSA algorithm to aggressively attenuate the band without structuring the noise to an annoying degree.

	Noise Masked	Noise Audible
Signal Masked	gain = 1	use old gain
Signal Audible	gain = 1	soften gain

Table 4.1 The four masking categories for each frequency bin when applying masking to the modified MMSE-LSA algorithm.

The final case is when both the signal and noise are audible. When both the noise and signal levels are substantially elevated above the masking threshold, there is no obvious means of making use of the masking threshold information. However, when the noise level is only slightly above the masking threshold so that the modified MMSE-LSA algorithm would attenuate the frequency band so that the noise level drops below the masking threshold, the applied gain can be "softened" so that the noise level is brought just below the masking threshold. The noise will be imperceptible just as if the gain were not softened, but the softened gain will lead to less signal distortion.

To judge the increase in performance of such a perceptually aware modified MMSE-LSA algorithm, the routine was run on the sentence "Why were you away a year Roy?" that had been contaminated with 10 dB car noise. We used such a completely voiced sentence so that the noise adaptation scheme would have an easier time segmenting the noisy speech, minimizing the impact of errors in distinguishing between the speech and noise. The speech was sampled at 8 kHz and 256-point frames with 50% overlap were used. All the frequency bins for every frame except the noise-only frames were first categorized according to signal and noise perceptibility. There were 124 such frames, and the bins were categorized as given in Table 4.2.

	Noise Masked	Noise Audible
Signal Masked	70.1%	21.1%
Signal Audible	6.5%	2.3%

Table 4.2 Relative frequency of masking categories for the DFT coefficients in the sentence "Why were you away a year Roy?" that had been contaminated with 10 dB car noise.

There is a large percentage of frequency bins where both the signal and noise are masked, which is due partly to inaccuracies in the masking model.

Figure 4.3 is a histogram of all the DFT coefficients corresponding to those frequency bins containing an audible signal but masked noise. They are grouped according to the gain assigned by the modified MMSE-LSA algorithm.

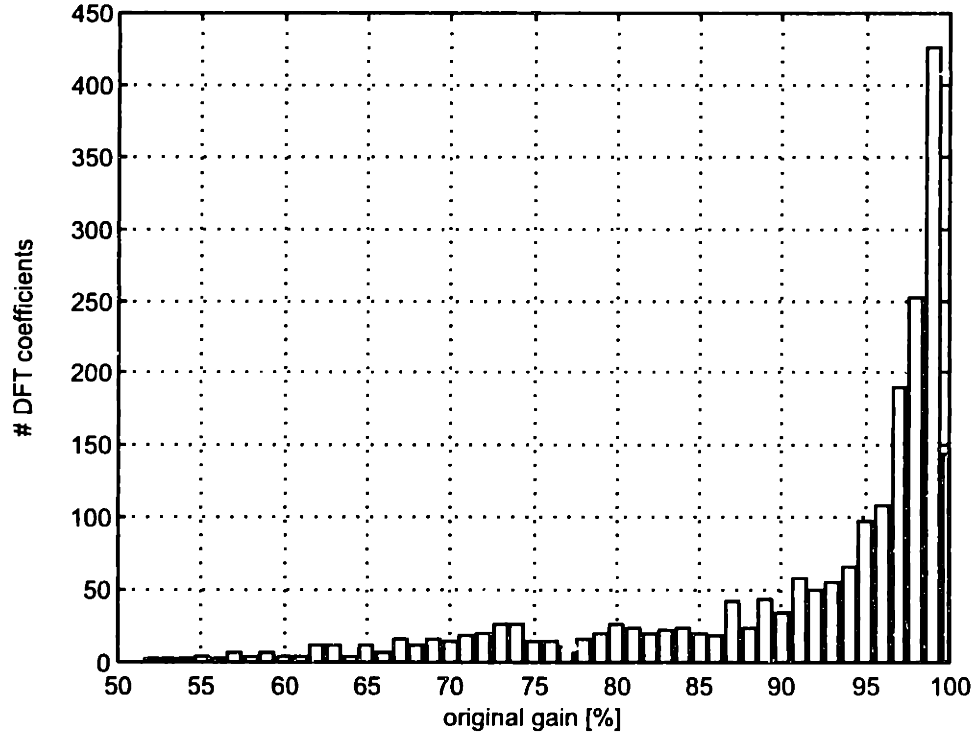


Figure 4.3 Histogram of DFT coefficients where the signal was audible but the noise masked, grouped according to the gain assigned by the modified MMSE-LSA algorithm.

As previously discussed, a unity gain should be applied to these coefficients since the noise is already inaudible to the human listener. However, the gains are already close to unity so the impact on the enhanced speech of identi'ying this case and modifying the gain is negligible.

Similarly, Figure 4.4 is a histogram of all the DFT coefficients corresponding to those frequency bins with both audible signal and noise. The coefficients are grouped according to the difference between the softened gain (the gain needed to bring the noise down to the masking threshold) and the gain assigned by the modified MMSE-LSA algorithm. We see that the modified MMSE-LSA gains are not much more aggressive than the softened gains.

Informal listening tests verified that the effect of softening the DFT coefficients was negligible on the algorithm's performance. We conclude that the modified MMSE-LSA algorithm naturally comes close to meeting these perceptual criteria without explicitly taking them into account.

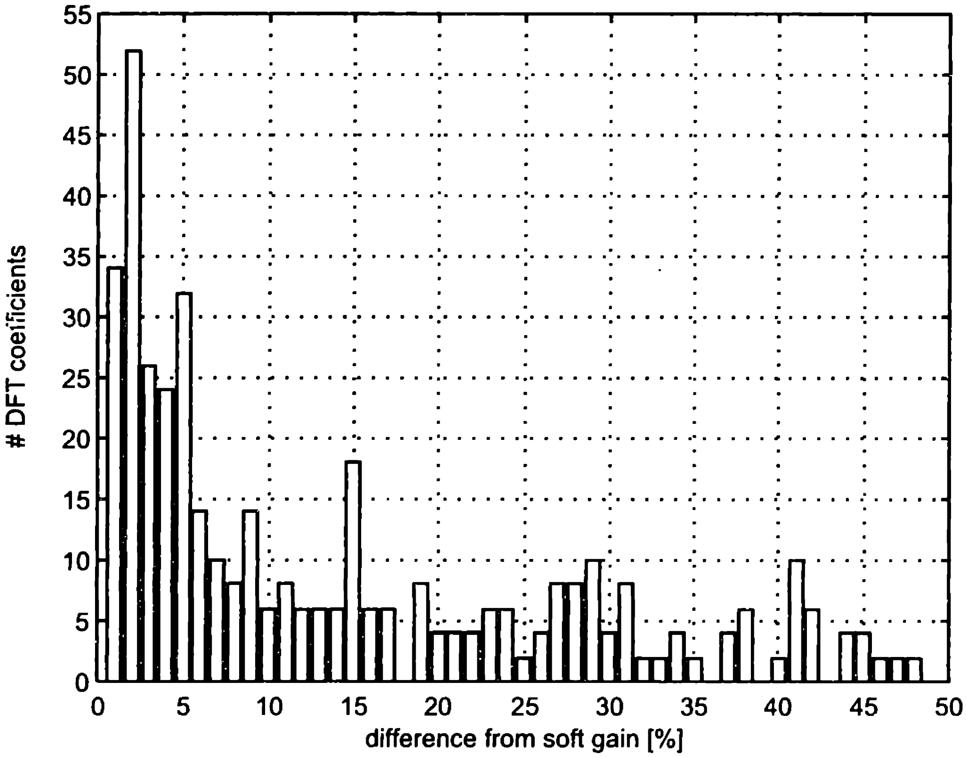


Figure 4.4 Histogram of DFT coefficients where both the signal and noise were audible, grouped by the difference between the softened gain and the modified MMSE-LSA gain.

Note that more elaborate perceptual modifications can be made with a loudness model. Such a model would give us the flexibility to modify the frequency gains on a continuous scale and might lead to a more substantial improvement in performance.

4.3 Application to the Signal Subspace Algorithm

Perceptual measures have a more natural application to the signal subspace algorithm, mainly because arbitrary spectral constraints on either the signal distortion or the noise energy are built into the algorithm. With a masking model, we can constrain the noise energy or the signal distortion to be inaudible by setting the α_k constants in (3.8) or (3.11) to an appropriate function of the masking threshold. For noisier environments and with the aid of a loudness model, more elaborate constraints can be concocted that could, for example, constrain the noise or distortion to be a certain number of sones louder that the clean speech.

Unfortunately, the errors in the perceptual measure calculations are magnified when the technique in Section 4.1 is applied in this manner. For example, constraining the residual noise energy to be below the masking threshold results in the noise being shaped so that it takes on the tonal characteristics of the clean speech. This effect is quite audible and annoying to the listener.

5

A Hybrid Algorithm

- *A Generalized Structure*
- *Signal Subspace as a Core Estimator*
- *Implementation for Listening Tests*
- *Results of Subjective Listening Tests*

5.1 A Generalized Structure

Ephraim's signal subspace approach (see Chapter 3) and Malah's modified MMSE-LSA algorithm (see Chapter 2) have very different strengths and weaknesses. The signal subspace algorithm provides a simple but theoretically elegant and powerful framework for trading off between the degree of noise suppression and signal distortion. This framework is general enough to incorporate many different criteria, including perceptual measures for general applications. This provides a good deal of flexibility when attempting to specialize an enhancement algorithm for a specific application. However, the technique offers no means for controlling noise distortion and handling non-stationary noise. Noise can be so severely distorted that the enhanced signal is less desirable than the original noisy signal, even though the noise energy has been suppressed. This forces one to operate the signal subspace algorithm in a very aggressive mode, so that the noise is practically eliminated but signal distortion may be high.

Malah's modified MMSE-LSA algorithm has been carefully designed to reduce noise distortion and adapt to non-stationary noise. The algorithm is quite robust when presented with different types and levels of noise. The main difficulty is that the tradeoff between the degree of noise suppression and signal distortion is awkward and is best performed by varying α in (2.16), which has undesirable side effects on the noise distortion. This provides very little flexibility when trying to adapt the algorithm to fit a particular application.

Our goal is to combine the strengths of these two methods in order to generate a robust and flexible speech enhancement algorithm that will exhibit better performance. Our approach is to analyze the general structure of Malah's modified MMSE-LSA scheme and generalize it.

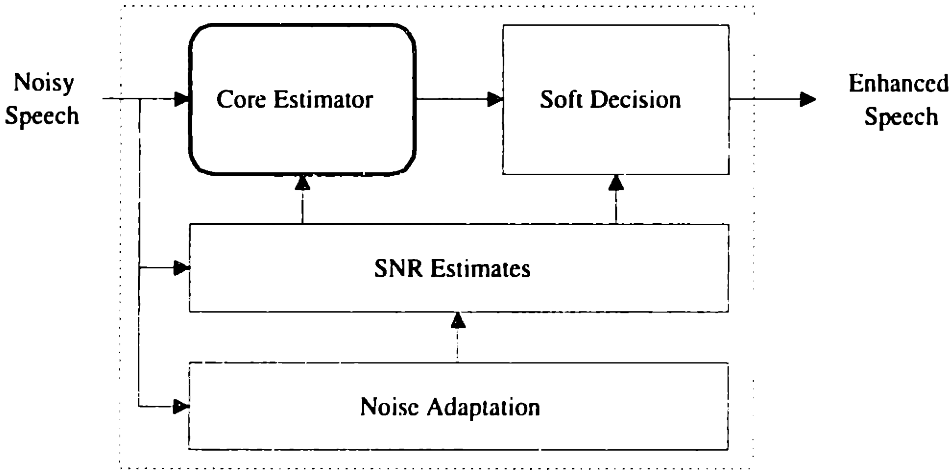


Figure 5.1 The general structure of the modified MMSE-LSA speech enhancement algorithm.

In Figure 5.1, the algorithm has been broken down by function. There is a "core estimator" (the MMSE-LSA in Malah's case) surrounded by some supporting structures. The noise adaptation scheme acts independently from the remainder of the modules. It is essential for many STSA speech enhancement algorithms to have an accurate estimate of the noise. Malah's implementation is particularly effective in tracking non-stationary noise, especially noise with varying intensity levels, and would be of use in any such STSA algorithm. The decision directed estimation approach is buried in the SNR estimator, which smoothes estimates between frames when the SNR becomes poor. We have seen that the effect is to reduce noise distortion when the gain applied depends heavily on these SNR estimates. The soft decision module has broad applicability, and could be considered part of the core estimator. Since this technique has proven most effective in handling the uncertainty of signal presence in certain frequency bands for different estimators, we consider the soft decision module to be a separate entity and require the core estimator to assume that the signal is always present in all frequency bands.

5.2 Signal Subspace as a Core Estimator

Our first insight is that we can substitute anything we desire in the "core estimator" block of Figure 5.1 and take advantage of the supporting structure as long as the effective gain depends heavily on the SNR estimates provided and the estimator does not take advantage of the uncertainty of signal presence. Our intuition is that this choice of core estimator might depend on the desired application. We will consider using different core estimators in Chapter 6. For our present purpose, however, we will use the spectral domain constrained version of the signal subspace algorithm from Chapter 2 as our core estimator in an effort to take advantage of the algorithm's aggressive noise suppression properties and flexibility.

We must first modify this algorithm so as to satisfy our constraints on the core estimator. Since the first step of the signal subspace algorithm is to decompose the noisy speech into a noise-only subspace and a speech-plus-noise subspace and throw away the noise-only subspace, the algorithm takes advantage of the uncertainty of signal presence. In fact, if the KLT is approximated with a DFT, this step is precisely a hard decision with zero gain applied to the frequency bins that contain pure noise. As discussed in Section 2.2, such an approach leads to unpleasant noise distortion properties. Our first modification then is to skip this subspace cancellation step.

Adapting the algorithm to be a function of our SNR estimates is a bit more troublesome. The first difficulty is that the signal subspace algorithm assumes the noise is white, and to be a function of SNRs for each frequency bin implies that the noise model must be generalized. We first make the Stationary Process Long Observation Time (SPLOT) assumption and approximate the KLT with the DFT.[26] We will now consider applying the signal subspace algorithm to a whitened version of the noisy speech. Say $\mathbf{W}$ is the whitening filter for the noise $\mathbf{w}$. Then, after applying $\mathbf{H}$ to the whitened noisy speech $\mathbf{Wy}$ we obtain an estimate of $\mathbf{Wx}$. Solving for $\hat{\mathbf{x}}$, we have

$$\hat{\mathbf{x}} = \mathbf{W}^{-1}\mathbf{H}\mathbf{W}\mathbf{y} \tag{5.1}$$

where

$$\mathbf{H} = \mathbf{U}\mathbf{Q}\mathbf{U}^{\#} \tag{5.2}$$

$$\mathbf{W} = \mathbf{U}\mathbf{W}_F\mathbf{U}^{\#} \tag{5.3}$$

Since we are using a DFT approximation to the KLT, $\mathbf{U}^{\#}$ is the DFT matrix operator and $\mathbf{U}$ is the inverse DFT matrix operator. In (5.3), $\mathbf{W}_F$ is the frequency domain implementation of the whitening filter. Therefore $\mathbf{W}_F$ is a diagonal matrix, and $\mathbf{Q}$ is diagonal as derived in Section 3.2. Substituting (5.2) and (5.3) into (5.1) and simplifying, we obtain

$$\begin{aligned}\hat{\mathbf{x}} &= \mathbf{U}\mathbf{W}_F^{-1}\mathbf{Q}\mathbf{W}_F\mathbf{U}^{\#}\mathbf{y} \\ &= \mathbf{U}\mathbf{Q}\mathbf{U}^{\#}\mathbf{y} \\ &= \mathbf{H}\mathbf{y}\end{aligned} \tag{5.4}$$

We have shown that whitening the signal, applying the signal subspace algorithm, and then applying the inverse of the whitening filter is equivalent to applying the signal subspace algorithm to the colored noise directly. However, there is a change in the form of the constraints.

For the whitened noisy input, we now have

$$E[|\mathbf{u}_k^{\#}\tilde{\mathbf{r}}_{\mathbf{w}}|^2] \le \alpha_k \tilde{\sigma}_{\mathbf{w}}^2 \tag{5.5}$$

where

$$\tilde{\mathbf{r}}_{\mathbf{w}} = \mathbf{H}\mathbf{W}\mathbf{w} \tag{5.6}$$

$$\tilde{\sigma}_{\mathbf{w}}^2 = E[|\mathbf{u}_k^{\#}\mathbf{W}\mathbf{w}|^2] \tag{5.7}$$

So $\tilde{\mathbf{r}}_{\mathbf{w}}$ given in (5.6) is the residual whitened noise, and $\tilde{\sigma}_{\mathbf{w}}^2$ given in (5.7) is the variance of this whitened noise. Since we are using the DFT approximation to the KLT, the expectations in (5.5) and (5.7) are energy spectral coefficients of the residual whitened noise and the whitened noise respectively. Therefore, dividing the k^{th} constraint given in (5.5) by the magnitude squared of the k^{th} component of the whitening filter in the frequency domain $|\mathbf{W}_{Fk}|^2$, we obtain our new constraint:

$$S_{\mathbf{r}_{\mathbf{w}}\mathbf{r}_{\mathbf{w}}}(k) \le \alpha_k S_{\mathbf{ww}}(k) \tag{5.8}$$

The final step is to choose the constant constraints α_k in (5.8). For white noise, Ephraim found that $\alpha_k = \exp\{-\upsilon\sigma_{\mathbf{w}}^2/\lambda_{\mathbf{x}}(k)\}$ was a good selection for aggressive noise suppression. For the DFT approximation to the KLT, we have $\lambda_x(k) = S_{xx}(k)$. Thus, to extend the technique to colored noise, it seems natural to try

$$\begin{aligned} \alpha_k &= \exp\{-\upsilon \cdot S_{ww}(k)/S_{xx}(k)\} \\ &= \exp\{-\upsilon/\xi_k\} \end{aligned} \tag{5.9}$$

In (5.9), we have ensured that the resulting gain depends heavily on the estimate of the a-priori SNR ξ_k, as required for a good core estimator.

The complete form of our new core estimator is shown in Figure 5.2, and the complete speech enhancement system using this core estimator shall be referred to as the "hybrid algorithm".

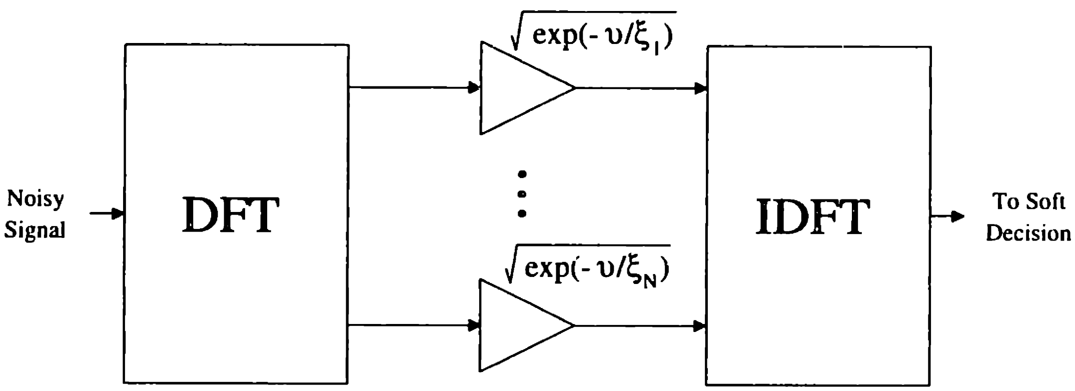

Noisy Signal
DFT
$\sqrt{\exp(-\upsilon/\xi_1)}$
$\sqrt{\exp(-\upsilon/\xi_N)}$
IDFT
To Soft Decision

Figure 5.2 Block diagram of an implementation of the core estimator of the hybrid algorithm.

5.3 Implementation for Listening Tests

We compared the performance of the hybrid algorithm with that of the modified MMSE-LSA and signal subspace algorithms, all at various levels of aggression. David Malah's original Matlab code from 1996 was used for the modified MMSE-LSA implementation. When used as a pre-processor for a speech coder, this version chose α based on the bit rate of the expected coder. We will consider Malah's 16 kbps (corresponding to $\alpha = 0.91$) and his more aggressive 8 kbps ($\alpha = 0.95$) settings. We modified Malah's code by replacing the MMSE-LSA estimator with the signal subspace core estimator derived above to obtain an implementation of the hybrid enhancement algorithm. There are two parameters that contribute to the aggression of the hybrid algorithm: α and υ. We used the same values of α as we did for the modified MMSE-LSA algorithm, and chose υ between 1 and 2 for increasing levels of aggression. All the algorithms in the test used 256-point frames with 50% overlap.

The signal subspace algorithm was implemented from scratch, in the form of the spectral domain constrained estimator. In order to make a fair comparison with the other enhancement algorithms, however, it was necessary to modify the spectral domain constrained estimator to handle non-stationary noise. We first consider the generalization of the algorithm to colored noise as described above, and use the exponential model for choosing α_k:

$$\alpha_k = \exp\{-\upsilon \cdot S_{ww}(k)/S_{xx}(k)\} \qquad (5.10)$$

We use Malah's noise adaptation scheme to estimate $S_{ww}(k)$, and obtain an estimate of $S_{xx}(k)$ through $S_{xx}(k) = S_{yy}(k) - S_{ww}(k)$. Now $S_{yy}(k)$ is estimated by analyzing the autocorrelation of the noisy speech. This is a standard approach to spectral estimation and is described in [27]. For each frame, we take 32 samples ahead and 128 behind in time and use this resulting superframe $s[n]$ of length $Q = 416$ for our analysis, as shown in Figure 5.3.

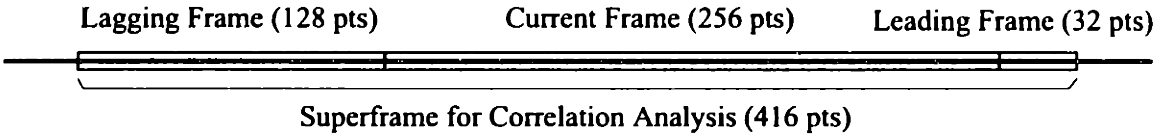


Figure 5.3 Composition of the superframe used for spectral analysis.

Next, we take the normalized autocorrelation of this sequence:

$$\hat{\phi}_{ss}[m] = \frac{1}{Q}\sum_{n=0}^{Q-|m|-1} s[n]s[n+|m|] \quad \text{for} \quad |m| \le Q-1 \qquad (5.11)$$

Finally, we window this autocorrelation with a Parzen window $p[n]$ of length 121 and take a DFT to obtain our spectrum estimate:

$$\hat{S}_{ss}[k] = \sum_{n=0}^{N-1} \hat{\phi}_{ss}[n] \cdot p[n] e^{-jn\frac{2\pi k}{N}} \qquad (5.12)$$

Taking a Fourier transform of the windowed autocorrelation sequence gives insight into the nature of this spectral estimate:

$$\hat{S}_{ss}(\omega) = \frac{1}{2\pi}\int_{-\pi}^{\pi} I(\theta) P(\omega - \theta) d\theta \qquad (5.13)$$

where $I(\omega)$ is the periodogram of $s[n]$ and $P(\omega)$ is the Fourier transform of the Parzen window $p[n]$. A Parzen window of length $2M + 1$ is given by

$$p[n]=\begin{cases} 1-\frac{6n^2}{M^2}\left(1-\frac{|n|}{M}\right) & 0\leq|n|\leq\lfloor M/2\rfloor \\ 2\left(1-\frac{|n|}{M}\right)^3 & \lfloor M/2\rfloor<|n|\leq M \end{cases} \tag{5.14}$$

and is shown in Figure 5.4 along with a Hamming and Rectangular window for comparison.[28]

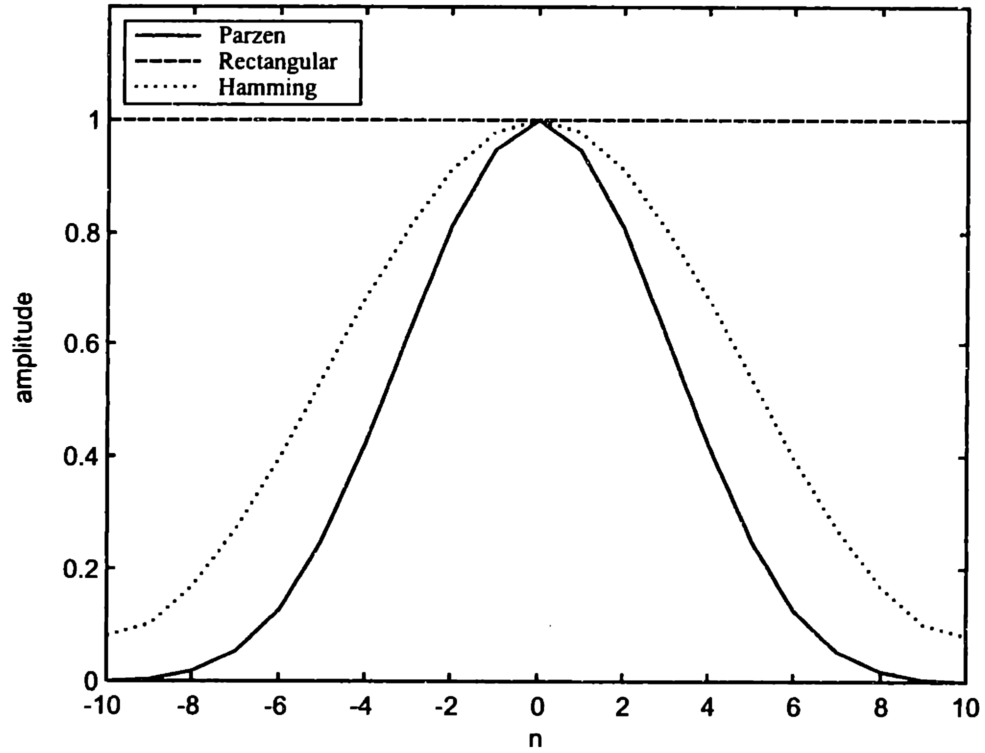


Figure 5.4 A comparison of 21-point Parzen, Rectangular, and Hamming windows in the time domain.

The normalized DFT of each of these windows in shown in Figure 5.5. We see that the Parzen window has the property that its transform is non-negative for all frequencies, implying from (5.13) that our spectral estimate will never be negative.

We can compute the expected value of our spectral estimate:

$$\begin{aligned} E[\hat{S}_{ss}(\omega)] &= \sum_{m=-(M-1)}^{M-1} E[\hat{\phi}_{ss}[m]]p[m]e^{-j\omega m} \\ &= \sum_{m=-(M-1)}^{M-1} \phi_{ss}[m]\left(\frac{Q-|m|}{Q}\right)p[m]e^{-j\omega m} \end{aligned} \tag{5.15}$$

In [27], it is shown that for $Q >> M$ and $p[0]=1$, the spectral estimate in (5.12) is asymptotically unbiased. The variance of $\hat{S}_{ss}(\omega)$ is shown to be

$$\mathrm{var}\{\hat{S}_{ss}(\omega)\} \approx \left(\frac{1}{Q} \sum_{m=-(M-1)}^{M-1} p^2[m] \right) S_{ss}^2(\omega) \tag{5.16}$$

From (5.16) we see that as the window length is decreased, the variance is reduced but at the expense of less frequency resolution. The window length must be chosen to balance between these two factors.

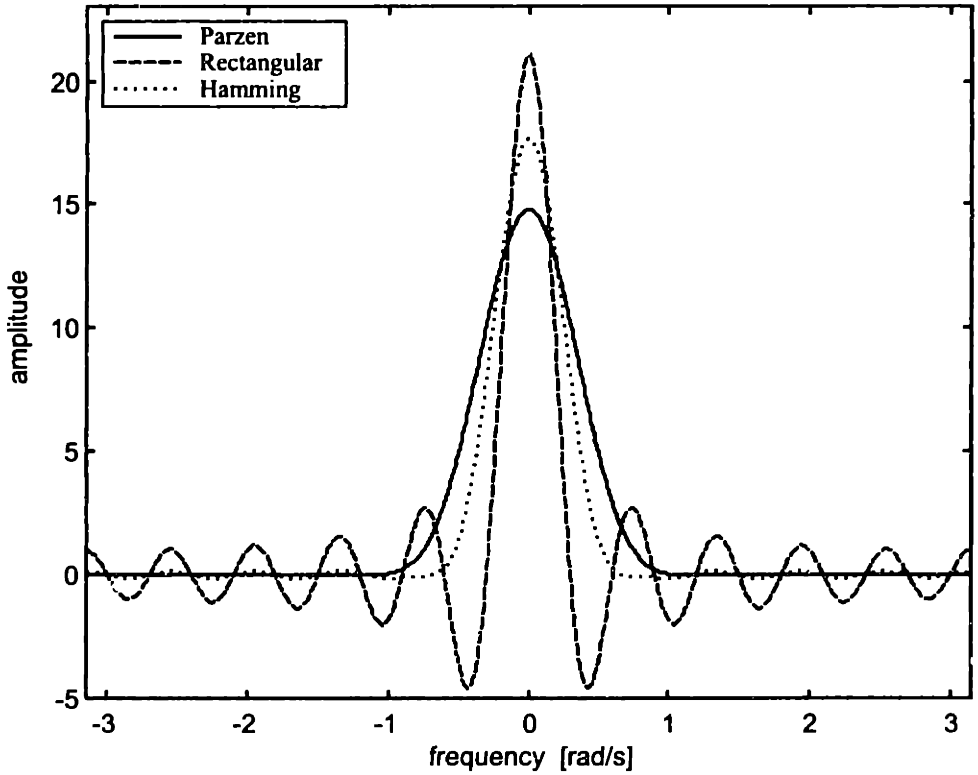


Figure 5.5 A comparison of 21-point Parzen, Rectangular, and Hamming windows in the frequency domain.

A different speech enhancement scheme recently developed by Motorola is used as a pre-processor for the IS-127 speech coder.[29] This enhancement scheme is part of the standard for the Enhanced Variable Rate Codec (EVRC) to be used in CDMA based telephone systems, and its structure is shown in Figure 5.6. The input speech is divided into 80-point frames. Each frame with 24 samples of the previous frame is multiplied by a smoothed trapezoidal window and transformed into the frequency domain via a 128-point DFT. These frequency components are grouped into 16 unequal bands and a gain is applied to each band before the signal is transformed back into the time domain and overlap-added to form the enhanced signal. The gain applied (in dB) is a piecewise-linear function of the estimated SNR in the channel, as given in Figure 5.7.

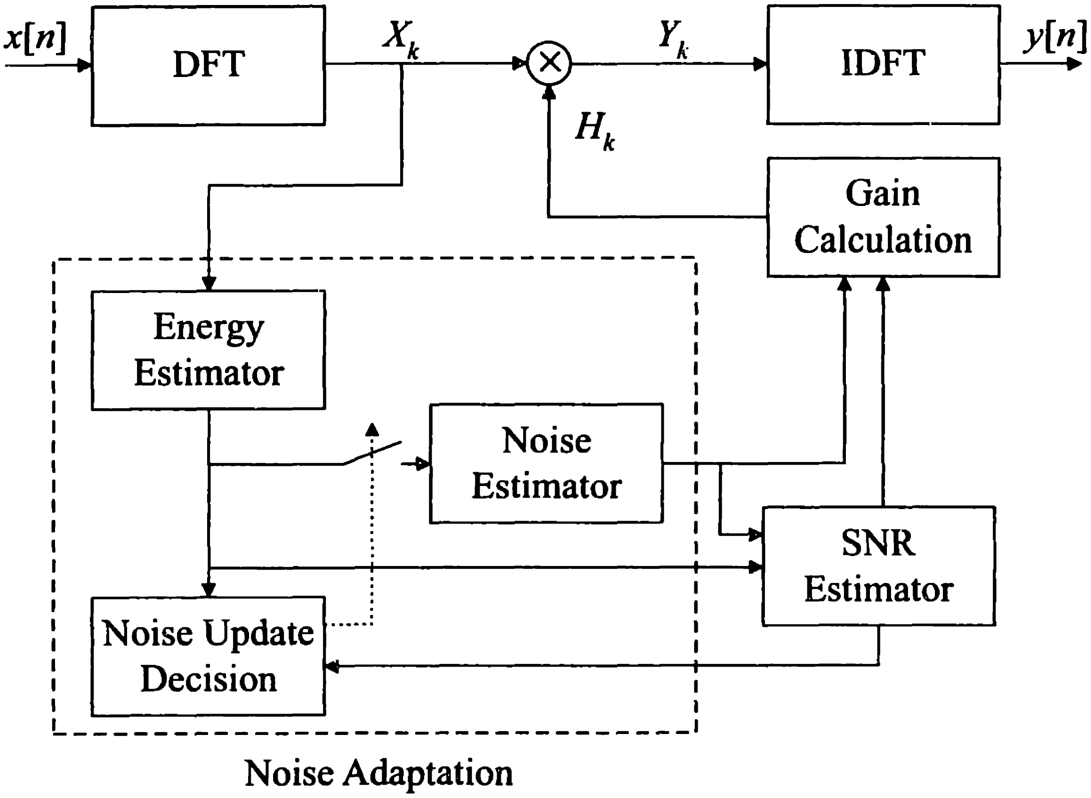


Figure 5.6 Block diagram of Motorola's speech enhancement system used as a pre-processor for the IS-127 speech coder.

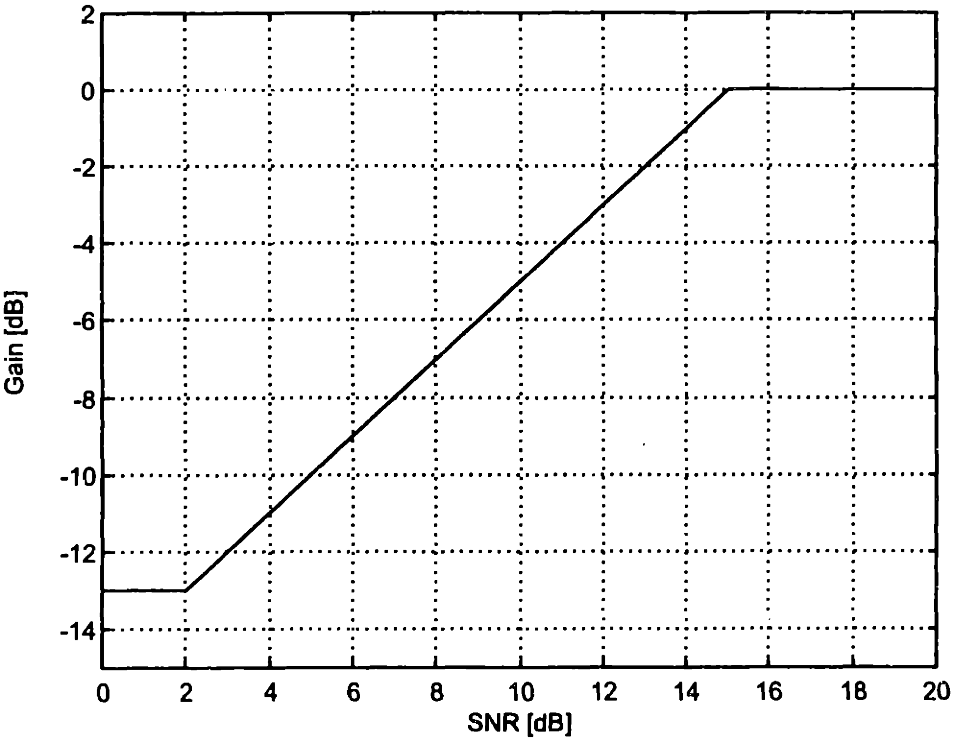


Figure 5.7 Gain as a function of SNR for Motorola's speech enhancement system.

The SNR is estimated by taking a ratio of the estimated channel energy and the estimated noise energy. The channel energy estimate is simply the actual energy of the current frame and is adaptive over frames with a tunable forgetting factor. The algorithm uses a spectral deviation measure from frame to frame to determine when to update the noise estimate. The algorithm's high-level structure is very similar to that of the modified MMSE-LSA algorithm, although the details are quite different.

5.4 Results for Subjective Listening Tests

Noisy speech samples enhanced with various settings for the hybrid, modified MMSE-LSA, signal subspace, and IS-127 algorithms were subjected to MOS listening tests after being coded by the IS-641 speech coder. The results of these tests are shown in Table 5.1. On the average over all pairs of conditions, MOS scores that differ by more than 0.16 are considered statistically different with a 95% level of confidence.

We first notice that our hypothesis in Section 2.4 that people favor Malah's 16 kbps setting over his 8 kbps setting, even when used as a pre-processor for a 7.4 kbps bit rate coder, is justified. However, here both of Malah's settings for the modified MMSE-LSA algorithm are preferred to the noisy speech, which was clearly not the case for his previous test (see Table 2.1). In this previous test, speech samples from a number of different speech coders that were necessarily of very different quality were presented to the subjects. In our recent test, nearly all speech samples were coded with IS-641. This difference in presentation could partially account for these dissimilar results.

We can also conclude that increasing the aggression level υ past 1 for the hybrid algorithm results in lower quality output, as the increase in distortion outweighs the additional reduction in noise level. Although it is not certain whether changing α from the 16 kbps setting to the 8 kbps setting makes much of a difference for low noise levels, it is clearly detrimental to quality for both 10 dB car noise and 10 dB babble.

The signal subspace algorithm, and to a lesser extent the hybrid algorithm, proved very destructive when enhancing babble. The difficulty is that some components of the babble are treated as speech by the noise adaptation scheme and SNR estimation due to shared spectral characteristics with the clean speech, and are not attenuated aggressively even though other components are. The result is a structuring of the noise that leads to annoying noise distortion. Both the hybrid and signal subspace algorithms perform well with car noise, with the hybrid algorithm providing better quality. The signal subspace algorithm is also the only technique that

	Clean	Babble 10dB	Babble 20dB	Car Noise 10dB	Car Noise 15dB	Car Noise 20dB
None	4.087	3.083	3.739	2.716	3.167	3.481
Hybrid 8/1	4.144	2.477	3.746	3.019	3.689	3.886
Hybrid 16/1	4.121	2.879	3.720	3.246	3.667	3.875
Hybrid 16/1.5	4.114	2.674	3.739	3.174	3.636	3.890
Hybrid 16/2	4.189	2.504	3.684	2.951	3.477	3.765
MMSE-LSA 16	4.117	3.213	3.787	3.186	3.583	3.788
MMSE-LSA 8	4.121	3.178	3.754	3.163	3.508	3.705
Subspace 10	3.773	2.538	3.648	2.883	3.383	3.750
IS-127	4.144	2.932	3.754	2.932	3.447	3.803

Table 5.1 MOS scores for different enhancement types (by row) and different noise types and intensities (by column). All speech samples were coded with IS-641 after enhancement. The aggression levels for each enhancement algorithm are also given, corresponding to Malah's 16 or 8 kbps setting and the value of υ, if applicable. For example, "Hybrid 16/1" is the hybrid algorithm with α set to correspond with Malah's 16 kbps tuning and with $\upsilon = 1$.

distorted the clean speech; the remainder of the enhancement algorithms improved the quality of the clean speech. This improvement is a common occurrence for speech enhancement tests since minor noise introduced in the recording of the original "clean" speech is removed by speech enhancement algorithms.

The IS-127 enhancement scheme is outperformed by the hybrid algorithm for car noise and by the modified MMSE-LSA algorithm for babble. However, the IS-127 scheme is quite robust to the different noise types and demonstrates a combined performance second only to the modified MMSE-LSA algorithm.

The hybrid algorithm performance was second to none when enhancing car noise, but introduced problems when dealing with babble. We would prefer an enhancement scheme that exhibits this level of performance with car noise, but also shows robustness with respect to noise types as the modified MMSE-LSA algorithm does. One means of achieving this is by using different core estimators and is described in Chapter 6.

6

Application: Speech Coder Pre-Processor

- *The IS-641 Speech Coder*
- *Specialized Pre-Processors*
- *Results with the Spectral Algorithm*

We will now investigate an application of the modular speech enhancement idea introduced in Section 5.1 to a specific example. We consider the IS-641 Algebraic Code Excited Linear Prediction (ACELP) speech coder, which has an LPC front-end. Recent speech enhancement schemes intended for use as pre-processors for speech coders are described in [30] and [29], where an emphasis is placed on the additional delay and complexity the speech enhancement contributes to the overall system. Our goal is to construct a pre-processor for the IS-641 that will improve the quality of the speech after being decoded. We will assume one additional liberty: that different speech enhancement algorithms may be used as pre-processors for different parameter extraction modules of the speech coder.

6.1 The IS-641 Speech Coder

The IS-641 speech coder is the enhanced full rate (EFR) speech codec that has been standardized in 1996 for the North American TDMA digital cellular system. This codec has been jointly developed by Nokia and the University of Sherbrooke.[31] The coder is based on the ACELP algorithm and has a bit rate of 7.4 kbps (neglecting error protection bits). The resulting speech quality is close to that of wireline telephony as compared with the G.726 32 kbps ADPCM speech coder, and is robust to errors arising from typical cellular operating conditions such as transmission errors, environmental noise, and the tandeming of speech coders.

The codec operates on 20 ms speech frames, which are each divided into four 5 ms subframes. A simplified block diagram is shown in Figure 6.1.

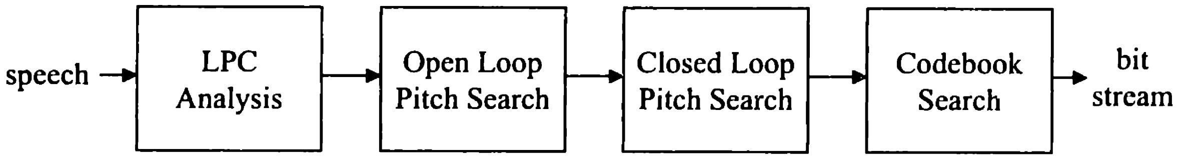


Figure 6.1 High-level block diagram of the IS-641 speech coder.

The input speech is first high-pass filtered and then a 10th order linear prediction analysis is carried out on the full frame. Recall from Section 1.3 that for our p^{th} order LPC model, the speech samples $s[n]$ are related to the excitation $u[n]$ by

$$s[n]=\sum_{k=1}^{p} a_k s[n-k]+Gu[n] \tag{6.1}$$

Our linear predictor with coefficients α_k is defined as a system with the output

$$\tilde{s}[n] = \sum_{k=1}^{p} \alpha_k s[n-k] \tag{6.2}$$

The prediction error $e[n]$ is defined as

$$\begin{aligned} e[n] &= s[n] - \tilde{s}[n] \\ &= s[n] - \sum_{k=1}^{p} \alpha_k s[n-k] \end{aligned} \tag{6.3}$$

Now we will solve for the parameters α_k that minimize the prediction error in a mean-square error sense over a short segment of the speech.[10] The short-time average prediction error is

$$\begin{aligned} E_n &= \sum_{m=0}^{N+p-1} e_n^2[m] \\ &= \sum_{m=0}^{N+p-1} \left(s_n[m] - \sum_{k=1}^{p} \alpha_k s_n[m-k] \right)^2 \end{aligned} \tag{6.4}$$

In (6.4), $s_n[m] = s[m+n] \cdot w[m]$ denotes a windowed segment of speech beginning at sample n. Say $w[m]$ can only be non-zero for $m = 0, \dots, N-1$. This windowed approach is known as the autocorrelation method for solving for the LPC coefficients, and is the method implemented by the IS-641 codec. We set $\partial E_n / \partial \alpha_i = 0$ in (6.4) and obtain the first order conditions

$$\sum_{k=1}^{p} \alpha_k \hat{\phi}_n[i,k] = \hat{\phi}_n[i,0] \qquad i = 1,2,\dots,p \tag{6.5}$$

where

$$\begin{aligned} \hat{\phi}_n[i,k] &= R_n[i-k] \\ &= \sum_{m=0}^{N-1-k+i} s_n[m] s_n[m+k-i] \end{aligned} \tag{6.6}$$

so from (6.5) we have the system of equations

$$\begin{bmatrix} R_n[0] & R_n[1] & R_n[2] & \cdots & R_n[p-1] \\ R_n[1] & R_n[0] & R_n[1] & \cdots & R_n[p-2] \\ R_n[2] & R_n[1] & R_n[0] & \cdots & R_n[p-3] \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ R_n[p-1] & R_n[p-2] & R_n[p-3] & \cdots & R_n[0] \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \vdots \\ \alpha_p \end{bmatrix} = \begin{bmatrix} R_n[1] \\ R_n[2] \\ R_n[3] \\ \vdots \\ R_n[p] \end{bmatrix} \tag{6.7}$$

The $p \times p$ matrix in (6.7) is Toeplitz, and therefore the parameters α_k can be obtained using the Levinson-Durbin recursion.[10] For robustness to quantization errors, these LPC coefficients are then converted into line spectrum pairs (LSP) and quantized using split vector quantization (SVQ).

The pitch analysis in the IS-641 codec is carried out in two stages. The first is an open-loop pitch search, where the correlation of a portion of the current frame with previous speech values is computed. The lags at which the maximum correlation occurs for each of three pitch period intervals are taken as pitch period candidates; one of them is then chosen based on the maxima, minima, and the degree of lag. Next, a closed-loop pitch search is performed on each subframe that estimates the pitch period with a fractional resolution of $^1/_3$.

Finally, algebraic codewords and corresponding gains are selected. The algebraic codebook structure is based on interleaved single-pulse permutation (ISPP) design. Here, each codeword vector contains 4 pulses, where each pulsc is restricted to a certain set of positions. The optimal pulse positions are calculated using a non-exhaustive analysis-by-synthesis search.

6.2 Specialized Pre-Processors

By enhancing the noisy speech before coding, we have already demonstrated that a dramatic improvement in quality can be obtained. When coded, noisy speech can produce annoying and unexpected artifacts since the codebooks of the coder are trained for quiet background conditions and the LPC model assumes clean speech. The effects of linear prediction analysis on noisy speech is well known. In [32], the Itakura distance measure was used to compare the LPC coefficients of the original clean speech with those of the noisy speech. The effects of quantization and white noise were studied, and the distortion due to white noise was determined to be serious.

In Figure 6.2, we attempt to improve the overall quality of the coded speech by using separate speech enhancement techniques as pre-processors for different components of the IS-641 coder. Specifically, we use one type of enhancement for the LPC analysis and another for the residual processing, including the pitch prediction and codeword selection procedures.

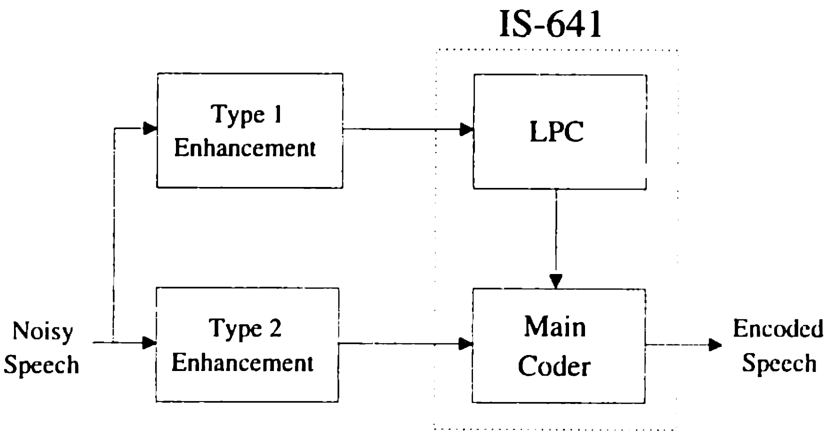


Figure 6.2 Using two different types of speech enhancement as a pre-processor for the IS-641 speech coder.

In the general case, using two different speech enhancement techniques will approximately double the complexity burden of implementing the algorithm. However, we will focus on a more specific case where each type of enhancement algorithm has Malah's framework shown in Figure 5.1, differing only in their core estimator structures. In this manner, both enhancement schemes can be run concurrently with a total complexity significantly less than that of running both separately.

6.3 Results with the Spectral Algorithm

The LPC analysis module of the IS-641 coder uses the correlation method described in Section 6.1 to compute the LPC coefficients. For this reason, it would seem plausible to use a core estimator that estimates the speech spectrum directly for the LPC analysis pre-processor. Under the assumption that the DFT coefficients of the noise and speech are zero-mean Gaussian random variables, we derive

$$\begin{aligned} E[|X_k|^2 | Y_k] &= \frac{S_{ww}[k] S_{xx}[k]}{S_{xx}[k] + S_{ww}[k]} + \left(\frac{S_{xx}[k]}{S_{xx}[k] + S_{ww}[k]} \right)^2 |Y_k|^2 \\ &= \frac{\xi_k}{1+\xi_k} \lambda_w + \left(\frac{\xi_k}{1+\xi_k} \right)^2 |Y_k|^2 \end{aligned} \tag{6.8}$$

and use this "spectral estimator" for our core estimator in the Type 1 Enhancement module. Note that this estimator depends heavily on the SNR estimates and does not take advantage of the uncertainty of signal presence, making it a valid core estimator. For our Type 2 Enhancement module, we shall use the hybrid algorithm developed in Chapter 5.

We first analyze how our scheme improves the LPC estimates of the IS-641 coder. We provide the pitch prediction and codebook modules of the coder (i.e., everything but the LPC analysis module) with the clean speech, and vary the type of enhancement used as a pre-processor for the LPC analysis. Then we can observe how well the LPC spectrum, defined as the frequency response of the filter

$$H(z) = \frac{G}{1 - \sum_{k=1}^{p} \alpha_k z^{-k}} \tag{6.9}$$

matches that of the clean speech. We compare the performance of the modified MMSE-LSA algorithm and the spectral algorithm (both set with an aggression level of 4 kbps) as well as that for no enhancement on a single frame of speech in Figure 6.3. Note how the noise attenuates and distorts the formants of the speech, and how it introduces an artificial formant around 2.4 kHz. The spectral algorithm does a much better job in tracking the original formants and in failing to reproduce this artificial formant. However, the performance of the spectral algorithm is far from perfect and some formant information is still clearly lost or distorted.

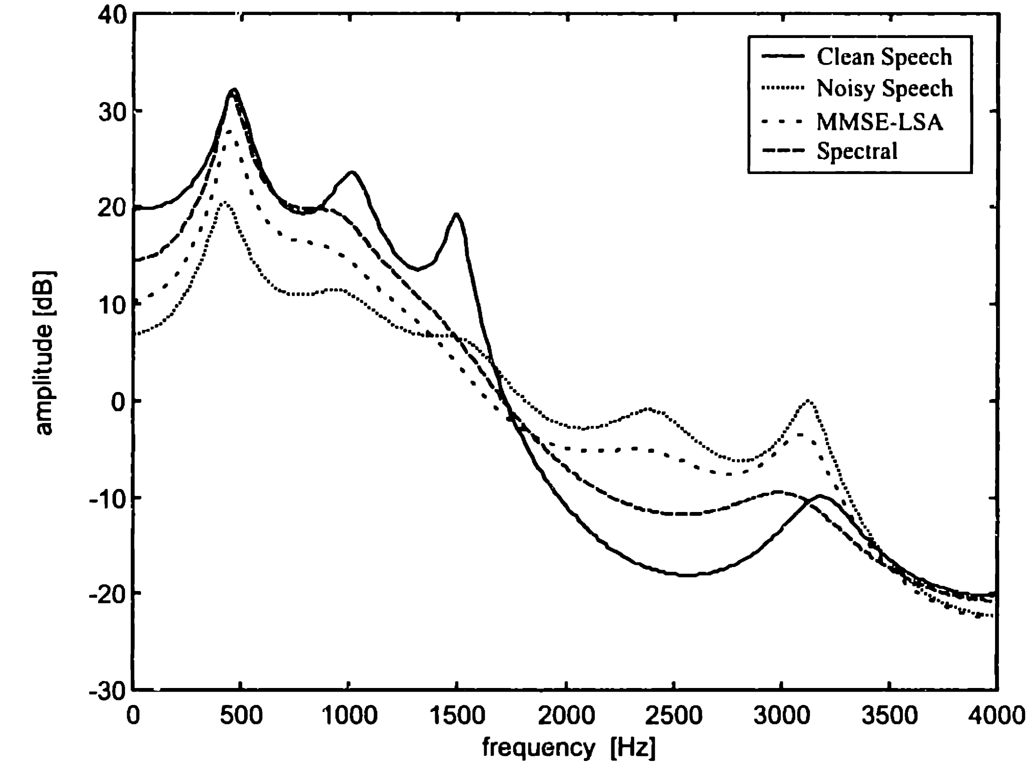


Figure 6.3 A comparison of LPC spectra for a frame of speech enhanced with the modified MMSE-LSA and spectral algorithms, with noisy and clean speech included as references.

The spectral algorithm does not out-perform the modified MMSE-LSA algorithm for all frames in this manner. There are many frames where both estimators are far off from the clean speech LPC spectrum.

Since our choice of Type 1 Enhancement affects the LPC spectrum used by the IS-641, we must also consider the impact of the spectral algorithm on the pitch estimation and codeword selection. Toward this end, we study the residual signal that is computed by filtering the input speech with $1/H(z)$, where $H(z)$ is the LPC filter in (6.9) calculated from the noisy speech after undergoing Type 1 Enhancement. Figure 6.4 compares the residual signals for a frame of speech where clean speech is used by the pitch prediction and codebook modules, but the input used for the LPC calculations is varied. We compare the performance of the spectral algorithm at both the

4 kbps and 8 kbps aggression levels with that of using no enhancement. The closed-loop pitch predictions for each of the 5 subframes are also denoted in the figure.

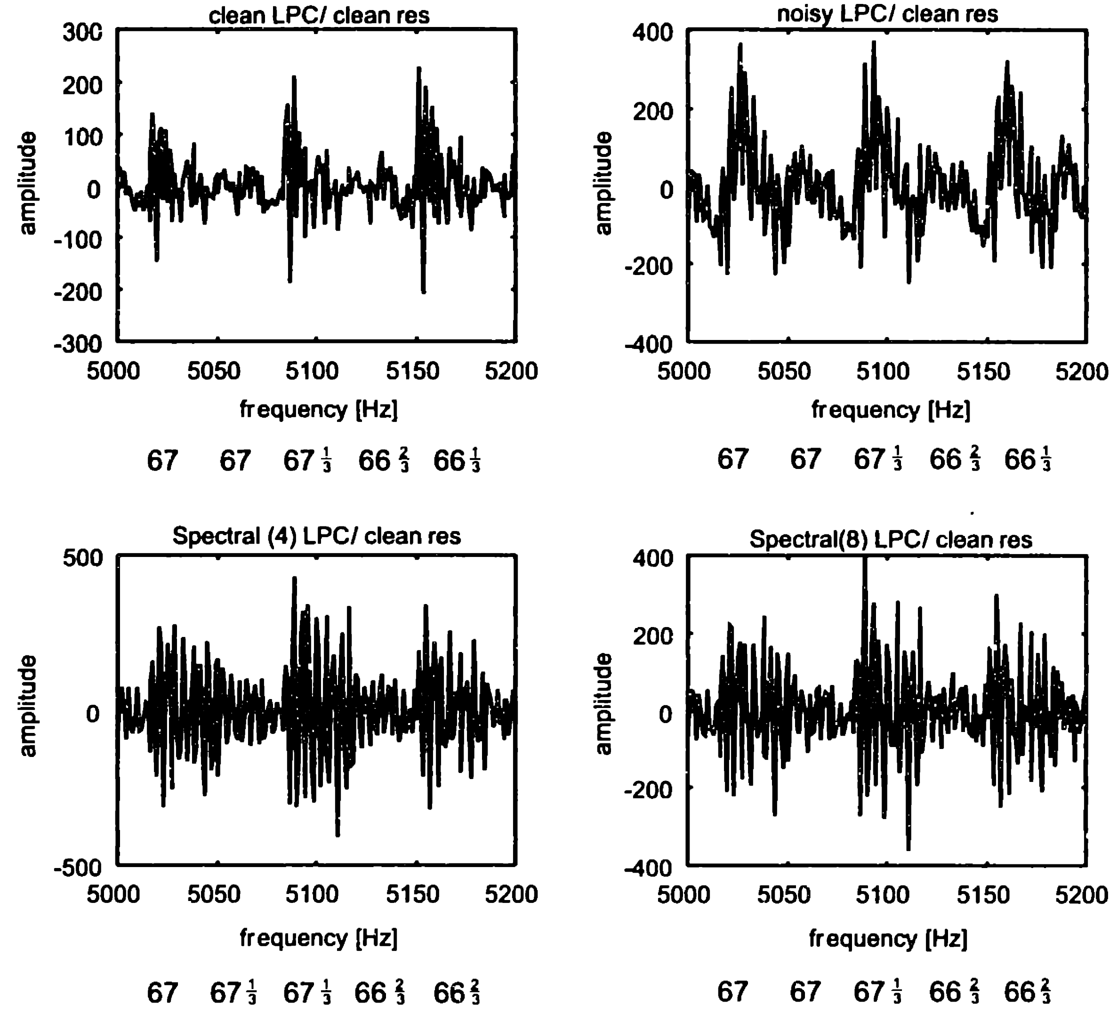


Figure 6.4 A comparison of LPC residuals computed using various inputs for the LPC analysis and the clean speech for the residual computation. Five subframes are shown. The closed-loop pitch estimate for each subframe is given below each graph.

We see that the choice of Type 1 Enhancement does not greatly impact the pitch prediction, as errors do not exceed more than $\frac{1}{3}$ of a sample. However, the form of the residual signals varies dramatically, suffering severe distortion when using the noisy speech and still a good deal of distortion when the spectral algorithm is used.

Finally, we consider the effect of Type 2 Enhancement on the residual signals. We supply clean speech for the LPC analysis and again compare the performance of the spectral algorithm at aggression factors of 4 kbps and 8 kbps with that of no enhancement. Here the impact on pitch prediction is more dramatic, as can be seen in Figure 6.5. The noisy speech causes severe pitch prediction distortions. When using the spectral algorithm, the pitch estimates are much more accurate, but at times errors are made where the pitch period is thought to be about half of the actual value. It is also very difficult to recover the structure of the clean residual with the spectral algorithm.

Note that the addition of wide-band noise gives rise to temporal spreading of the residual energy, and so processing these residuals to make them more "pulse-like" is an effective means of noise suppression.[33]

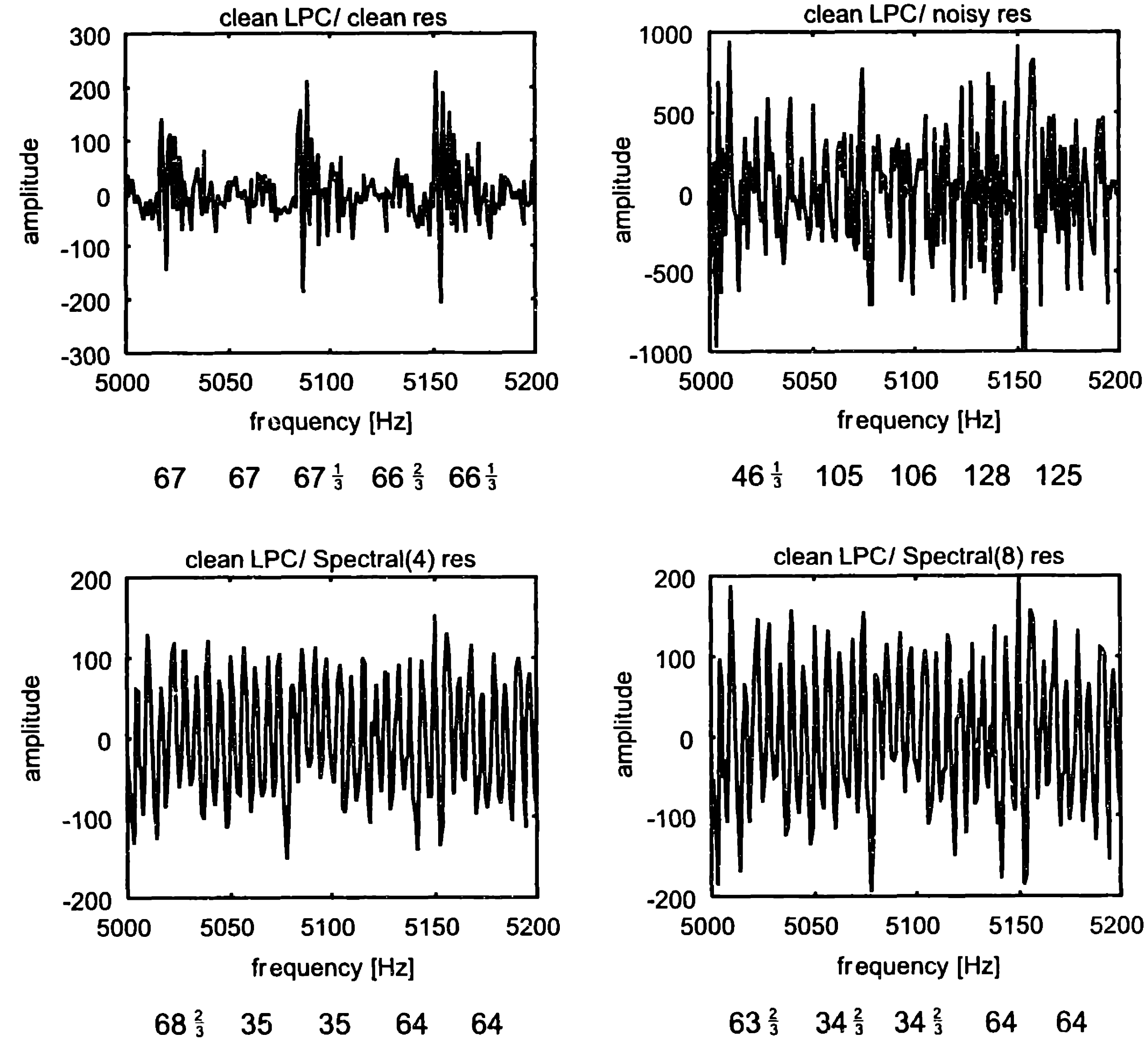


Figure 6.5 A comparison of LPC residuals computed using clean speech for the LPC analysis and various inputs for the residual computation. Five subframes are shown. The closed-loop pitch estimate for each subframe is given below each graph.

From informal listening, we found that when using some form of enhancement the pitch prediction errors were not terribly annoying in the decoded speech, manifesting themselves as subtle clicks in the worst cases. The LPC spectrum distortion and residual distortion were much more serious. We determined that the hybrid algorithm was a good choice as a Type 2 Enhancement method, since it provided flexibility over the aggression level and could have a substantial impact on residual noise reduction. We used the spectral algorithm at a low aggression setting (8 kbps or 16 kbps) for the Type 1 Enhancement, as we noted that less aggression at this stage produced decoded speech that was less "muffled" and not noticeably more noisy. Unfortunately, while the choice of Type 1 Enhancement makes a big difference in perceptual quality when the remainder of the coder has access to the clean speech, the difference when a noisy or enhanced speech signal is used is much less prominent.

MOS test results are shown in Table 6.1. This data contains that in Table 5.1, with the addition of the spectral algorithm and two "combination" enhancement schemes that use the spectral algorithm for LPC analysis and the hybrid algorithm for the residual computation.

	Clean	Babble 10dB	Babble 20dB	Car Noise 10dB	Car Noise 15dB	Car Noise 20dB
None	4.087	3.083	3.739	2.716	3.167	3.481
Hybrid 8/1	4.144	2.477	3.746	3.019	3.689	3.886
Hybrid 16/1	4.121	2.879	3.720	3.246	3.667	3.875
Hybrid 16/1.5	4.114	2.674	3.739	3.174	3.636	3.890
Hybrid 16/2	4.189	2.504	3.684	2.951	3.477	3.765
MMSE-LSA 16	4.117	3.213	3.787	3.186	3.583	3.788
MMSE-LSA 8	4.121	3.178	3.754	3.163	3.508	3.705
Subspace 10	3.773	2.538	3.648	2.883	3.383	3.750
Spectral 16	4.125	3.197	3.852	3.152	3.529	3.750
Spectral 16, Hybrid 16/1	4.098	2.928	3.837	3.216	3.655	3.928
Spectral 16, Hybrid 16/2	4.144	2.545	3.739	3.015	3.583	3.924
IS-127	4.144	2.932	3.754	2.932	3.447	3.803

Table 6.1 MOS scores for different enhancement types (by row) and different noise types and intensities (by column). All speech samples were coded with IS-641 after or during enhancement. The aggression levels for each enhancement algorithm are also given, corresponding to Malah's 16 or 8 kbps setting and the value of v, if applicable. For example, "Spectral 16, Hybrid 16/1" is the result of using the spectral algorithm with Malah's 16 kbps tuning for LPC analysis and the hybrid algorithm with α set to correspond with Malah's 16 kbps setting and with $v = 1$ for the residual computation.

We again notice that the more aggressive enhancement scheme, using the hybrid algorithm with $\upsilon = 2$ for the residual computation, is not preferred to the less aggressive one. However, increasing υ for the pure hybrid algorithm has much more of a damaging effect on quality for car noise than the same increase when the hybrid algorithm is used for combined enhancement. The combination enhancement technique exhibits similar or better performance than both the hybrid and spectral algorithms in car noise for all tested noise and aggression levels, with the exception of cases where the spectral algorithm is preferred to combination enhancement with $\upsilon = 2$ for 10 dB car noise.

Additionally, the spectral algorithm is the only single enhancement technique that significantly improves speech contaminated with 20 dB babble. This performance is also exhibited by the combination enhancement with $\upsilon = 1$, but is unfortunately not passed on for 10 dB babble. This is due to the severe distortion introduced by the hybrid algorithm for the residual calculation. Although these effects are relatively small, the data indicates that a combination enhancement approach may lead to a more robust system that shares the benefits and shortcomings of each constituent enhancement scheme.

7

Dual Channel Background

- *A Linear Model*
- *Separation by Decorrelation*
- *A Parametric Formulation*

A natural extension of the traditional speech enhancement paradigm would allow for multiple observations of the noisy speech signal. For example, Figure 7.1 illustrates a simplified application scenario depicting two microphones observing two competing sources in a room. There is a primary microphone (Microphone 1) that is positioned close to the speech source, and therefore measures a signal that consists mostly of speech with some noise contamination. Similarly, a secondary microphone (Microphone 2) is positioned much closer to the noise source than the speech, and measures mostly noise that is contaminated by some speech. The idea is that the secondary microphone can be used as a reference to cancel the noise in the primary microphone. There is a tradeoff between noise reduction and distortion inherent in such a system, similar to that for single channel enhancement.[34]

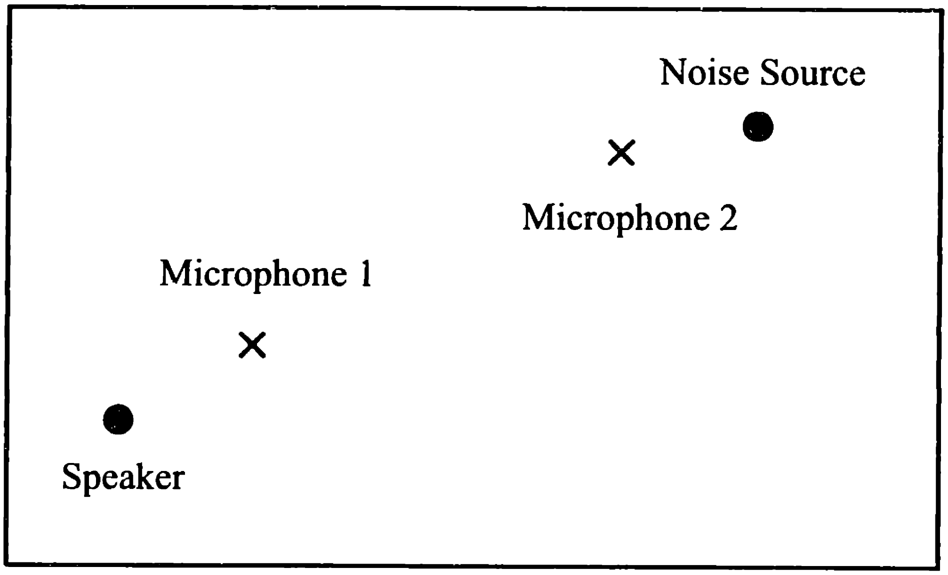


Figure 7.1 Simplified 2-source, 2-microphone speech enhancement scheme in a rectangular room.

7.1 A Linear Model

We model the dual speech enhancement scheme with the discrete time, LTI, multiple input multiple output (MIMO) system depicted in Figure 7.2. Here the acoustic transmission systems between each source and microphone are modeled as single discrete time transfer functions. We assume discrete time representations for all the physical signals involved in order to simplify the analysis of the digital reconstruction system. The goal is to estimate $s[n]$ given $y_1[n]$ and $y_2[n]$.

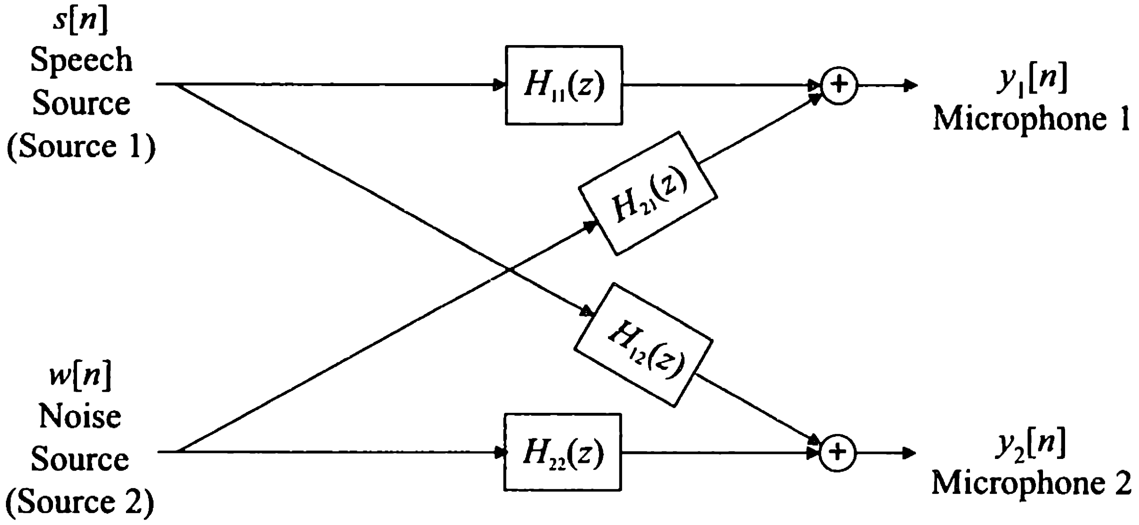


Figure 7.2 A discrete-time linear model for 2-source, 2-microphone speech enhancement.

In Figure 7.2, $H_{xy}(z)$ models the acoustic transfer function between source x and microphone y. For arbitrary transfer functions $H_{xy}(z)$, the reconstruction is not well defined since there are too many degrees of freedom in the problem. For example, say $s[n]$ and $w[n]$ are given and we then choose a particular set of $H_{xy}(z)$ transfer functions, which fully determines $y_1[n]$ and $y_2[n]$. Now, we can modify $H_{21}(z)$ in some manner and, under certain assumptions regarding the characteristics of $s[n]$ and $w[n]$, we can compensate for the resulting changes in $y_1[n]$ by making certain modifications to $H_{11}(z)$. Thus, we have the same input and output signals but a different MIMO system.

Clearly, we must constrain the problem further. A common assumption in the literature is to assume unity systems for $H_{11}(z)$ and $H_{22}(z)$, and FIR filter models for $H_{12}(z)$ and $H_{21}(z)$.[35] This is justified by considering applications where the microphones are very close to their corresponding sources and by using sufficiently high FIR filter orders so as to accurately model the room acoustics.

The transfer function for this MIMO system is given by

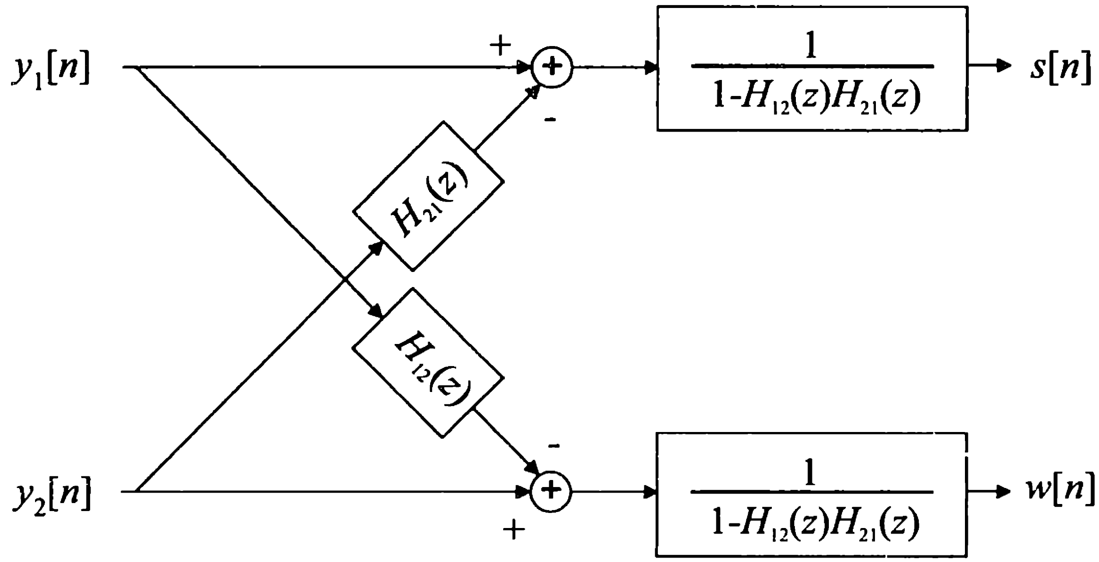
$$\begin{bmatrix} Y_1(\omega) \\ Y_2(\omega) \end{bmatrix} = \mathbf{H}(\omega) \cdot \begin{bmatrix} S(\omega) \\ W(\omega) \end{bmatrix} \qquad (7.1)$$

$$\mathbf{H}(\omega) = \begin{bmatrix} 1 & H_{21}(\omega) \\ H_{12}(\omega) & 1 \end{bmatrix} \qquad (7.2)$$

Inverting $\mathbf{H}(\omega)$ in (7.2), we have

$$\mathbf{H}^{-1}(\omega) = \frac{1}{1 - H_{21}(\omega)H_{12}(\omega)}\begin{bmatrix} 1 & -H_{21}(\omega) \\ -H_{12}(\omega) & 1 \end{bmatrix} \tag{7.3}$$

A direct implementation of the inverse in (7.3) is shown in Figure 7.3. This reconstruction system first involves the subtraction of the noise estimate from the secondary microphone, followed by a filtering operation that attempts to remove reverberations and other signal distortion from the speech.



$y_1[n]$
$y_2[n]$
+
-
-
+
$H_{21}(z)$
$H_{12}(z)$
$\frac{1}{1-H_{12}(z)H_{21}(z)}$
$\frac{1}{1-H_{12}(z)H_{21}(z)}$
$s[n]$
$w[n]$

Figure 7.3 An implementation of the reconstruction system for 2-source, 2-microphone speech enhancement.

A natural technique for reconstructing the original speech and noise signals is to estimate the transfer functions $H_{12}(\omega)$, $H_{21}(\omega)$ and then apply the reconstruction system in Figure 7.3 to the microphone outputs, with the system estimates substituted for the actual systems.

7.2 Separation by Decorrelation

We have assumed that the clean speech and the noise are uncorrelated signals. A key idea in recent work regarding the more general problem, where $s[n]$ and $w[n]$ are not restricted to speech and noise but may be arbitrary signals, is to make this correlation assumption a criterion for estimation.[35] That is, we will constrain our estimates of $H_{12}(z)$ and $H_{21}(z)$ so that the outputs $s[n]$ and $w[n]$ are uncorrelated. Now, given one of $H_{12}(z)$ or $H_{21}(z)$, the other system is fully determined via the auto-spectra and cross-spectra of the microphone signals. By relating the auto-spectra and cross-spectra of the outputs and inputs of the reconstruction system, setting $S_{\hat{s}\hat{w}}(\omega) = 0$ for the decorrelation condition, and solving for $\hat{H}_{21}(\omega)$, we obtain

$$\hat{H}_{21}(\omega) = \frac{S_{y_1 y_2}(\omega) - \hat{H}^*_{12}(\omega) S_{y_1 y_1}(\omega)}{S_{y_2 y_2}(\omega) - \hat{H}^*_{12}(\omega) S_{y_2 y_1}(\omega)} \qquad (7.4)$$

It was pointed out in [35] that for the special case where $\hat{H}_{12}(\omega) = 0$, (7.4) reduces to

$$\hat{H}_{21}(\omega) = \frac{S_{y_1 y_2}(\omega)}{S_{y_2 y_2}(\omega)} \qquad (7.5)$$

This is the classical least-squares solution where the secondary source (the reference signal) is assumed to be known.[36] Thus, the decorrelation condition is a generalization of Widrow's least-squares technique.

More constraints are needed in order to obtain a unique solution, since (7.4) only provides one equation for our two unknown systems. We assume that the systems are FIR, which reduces the possible solution set. One algorithm that implements this decorrelation condition is to alternate between estimating each set of filter taps given the other filter. So, the taps of $H_{21}(z)$ will be updated given $H_{12}(z)$, and vice-versa. The result is a sequential and adaptive algorithm that yields good results and convergence properties for low order models.[35] The difficulty in using this algorithm for speech enhancement is that as the number of FIR coefficients increases, the solutions become more and more ill-conditioned. To effectively model the room acoustics depicted in the simple setup in Figure 7.1, hundreds or even thousands of FIR filter taps may be required.[37]

7.3 A Parametric Formulation

A more tractable solution to the dual channel speech enhancement problem can be obtained by making more assumptions on the speech and noise characteristics. We can model the speech as a Gaussian AR process, and assume white Gaussian noise and FIR filter models for $H_{21}(z)$ and $H_{12}(z)$. In [38], all these parameters are estimated using a maximum-likelihood criterion. Since the likelihood functions are determined by unknown information, we can compute and use their expectations instead, and then apply the Estimate-Maximize (EM) algorithm. A computationally efficient sequential/adaptive algorithm based on this idea is described in [38], which is an extension of previous frequency domain techniques from [39].

8

A Dual Channel Application

- *Assumptions and Experimental Setup*
- *System Identification Review*
- *Experimental Results*

We will describe a dual channel speech enhancement application and cast the problem in the framework outlined in Chapter 7. After reviewing relevant system identification techniques, we attempt to characterize the performance of such a speech enhancement system.

8.1 Assumptions and Experimental Setup

We consider an application to a cell phone, where we wish to use speech enhancement as a pre-processor. The user speaks into the cell phone and the outgoing speech is enhanced before being coded and transmitted. The major limitation is that all the speech enhancement hardware must physically reside on the phone. It was suggested that a microphone array situated on the phone would be able to collect more information about the noise characteristics than a single microphone placed at the mouthpiece could, and would therefore lead to a more effective pre-processor than a single channel enhancement technique.

In order to evaluate this hypothesis and explore a multi-microphone solution to the application problem, we consider the two-microphone setup illustrated in Figure 8.1.

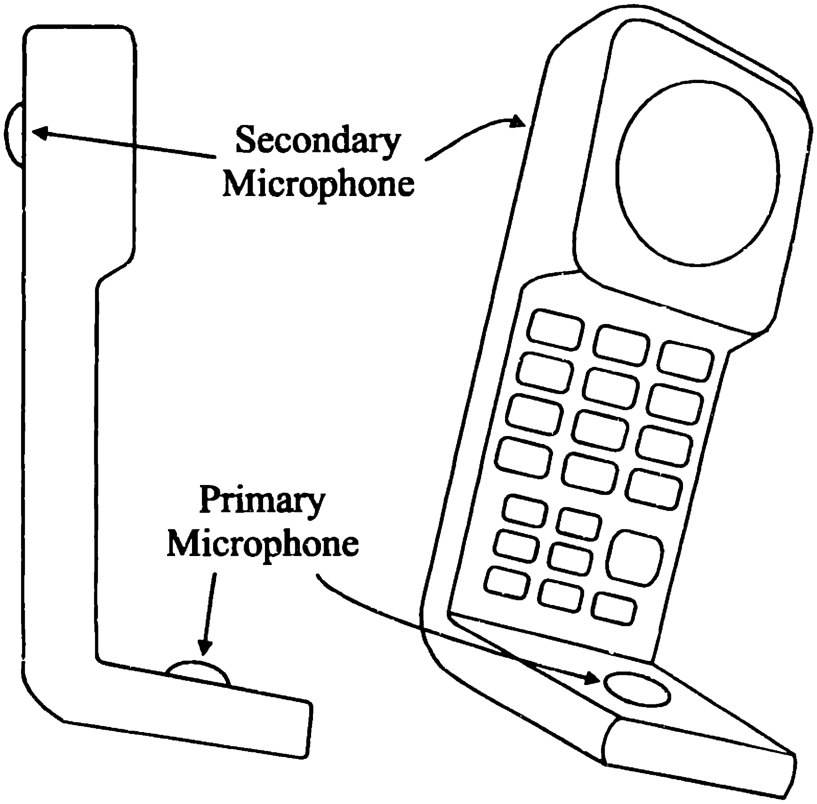


Figure 8.1 A two-microphone configuration intended for speech enhancement on dialogue during on a cell phone call.

In this scenario we have two microphones. The primary (or speech) microphone is situated on the mouthpiece where it is typically located; its purpose is to record with the best SNR possible and therefore provide information about the speech. The secondary (or noise) microphone is placed on the backside of the handset, as far away from the primary microphone as practically possible, with the intent of recording with the worst SNR in order to provide information about the noise. Since the intensity of the speech sound waves drops off as $1/r^2$, where r is the distance away from the speech source, the intention of this setup is to take advantage of the significantly greater speech strength at the primary microphone. We assume that the noise source(s) is sufficiently far away so that the intensity of the noise is, to first order (i.e. ignoring head diffraction effects). approximately the same at each microphone. In the limiting case where the noise source is at the same location as the speech source (and directionality effects are ignored), no advantage is gained by using more than one microphone for enhancement.

An experiment was conducted as shown in Figure 8.2.

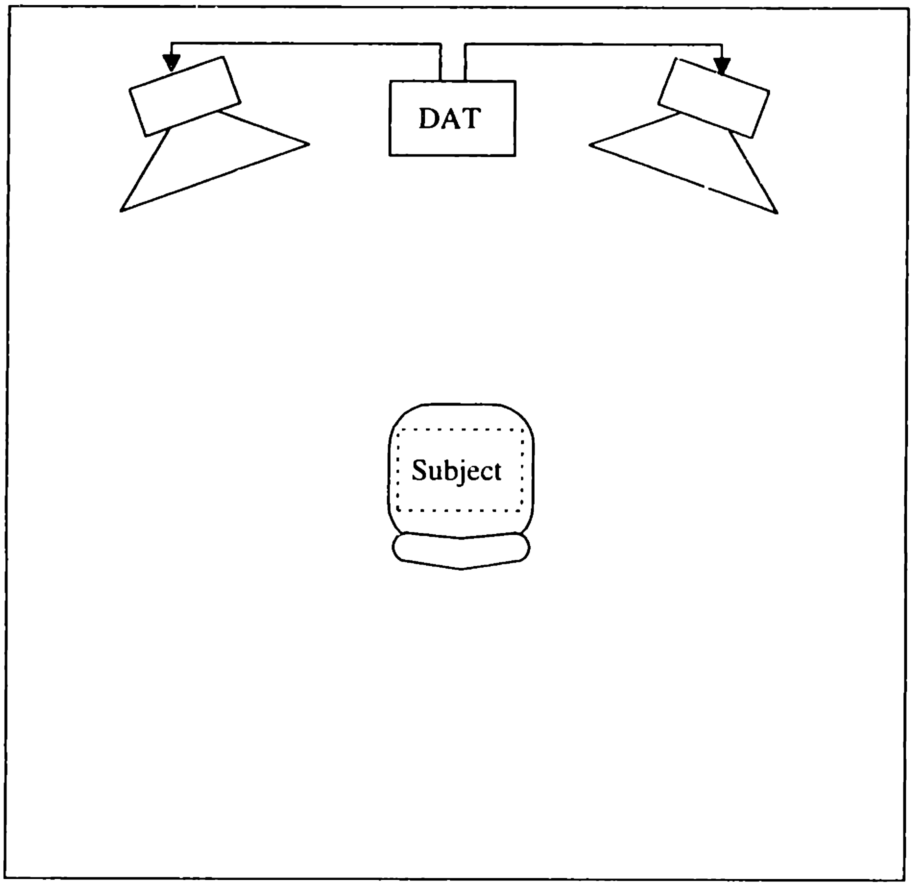


Figure 8.2 Experimental setup consisting of two noise sources and a human subject holding a cell phone with two microphones. Both noise sources are driven with the same noise data from a DAT player.

The experiment was carried out in a listening room at AT&T, where the ceiling and wall acoustic reflections can be considered negligible. Pre-recorded car noise and babble were played from a DAT tape through two speakers at one wall of the room, while a subject sat in the center of the room and faced the wall. The subjects spoke a number of sentences into a cell phone held in their right hand that had been prepared with two microphones as shown in Figure 8.1. The microphone outputs were recorded onto another DAT and later analyzed. The subjects were instructed to hold the phone as steady as possible in order to minimize the nonstationarity of the acoustic systems under consideration.

Now we apply the linear model developed in Section 7.1. Since we cannot assume that the forward acoustic systems $H_{11}(z)$ and $H_{22}(z)$ are unity systems as is frequently done in the literature, we first factor these systems so the overall MIMO system is separated into two subsections as shown in Figure 8.3. We will be content with reconstructing $s_0[n]$ from the microphone signals, so we only need to consider the right-hand portion of the flow graph.

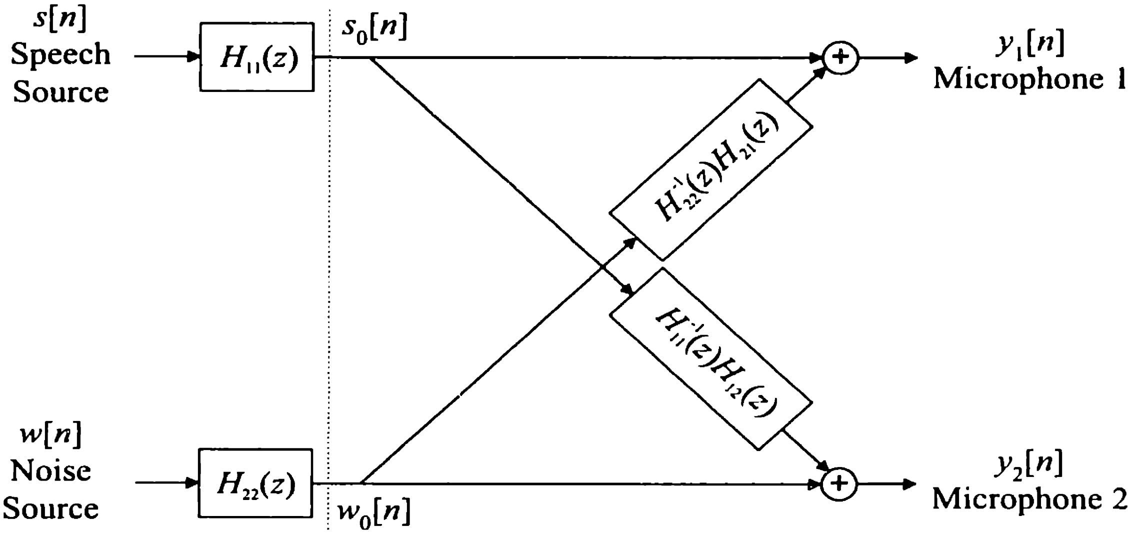


Figure 8.3 Factoring the acoustic MIMO system to reveal a subsection with unity forward acoustic systems.

Now we have unity forward acoustic systems, and cross systems that we will denote as

$$G_{12}(z) = H_{11}^{-1}(z)H_{12}(z) \tag{8.1}$$

and

$$G_{21}(z) = H_{22}^{-1}(z)H_{21}(z) \tag{8.2}$$

An important note is in order at this point. Since the inverse of this MIMO system involves the estimation of $G_{12}(z)$ and $G_{21}(z)$, we require that these systems be causal or causal after applying a finite delay to the input. Excluding unlikely pole-zero cancellations, this implies from (8.1) and (8.2) that $H_{11}^{-1}(z)$ and $H_{22}^{-1}(z)$ be causal after some finite delay so that the systems $H_{11}(z)$ and $H_{22}(z)$ must be minimum phase systems composed with a finite delay. However, these are acoustic systems and such systems need not satisfy this requirement, which is especially likely when inside a room where the source is significantly far away from the receiver.[40] Therefore, perfect reconstruction of the speech $s_0[n]$ is impossible for some acoustic environments. To make matters worse, attempting to invert such nonminimum phase systems with the inverse of the minimum phase component produces tonal artifacts, which would make for a terrible speech enhancement system as we learned in the single channel case. However, this drawback is no reason to abandon the multi-microphone solution. Nonminimum phase systems are easily identified, and the enhancement system can revert to a single microphone enhancement operation when such conditions are encountered. There is still the prospect of improved performance when such difficulties do not arise.

8.2 System Identification Review

Given an input $x[n]$ to a system and an output $y[n]$, we wish to be able to express $y[n] = g[n] * x[n]$, where $g[n]$ is a linear time-invariant (LTI) system. The art of estimating $g[n]$ given both $x[n]$ and $y[n]$ is part of the field of system identification.[41] The first step when dealing with such a problem is to model the system and account for any modeling errors. A useful representation is

$$y[n] = g[n] * x[n] + h[n] * e[n] \tag{8.3}$$

In (8.3), $h[n] * e[n]$ represents the effects of signals beyond our control that also affect the system, such as measurement errors and uncontrollable inputs. We assume that $e[n]$ is a sequence of independent, identically distributed (IID) random variables with a certain probability density function, and that $h[n]$ is an LTI system. This restriction does not allow for a completely general probabilistic model of disturbances, but is flexible enough for most practical problems.

We further simplify the model by assuming certain structures for the transfer functions $G(z)$ and $H(z)$. The simplest model used is called the ARX model, where these systems have the forms

$$G(z) = \frac{B(z)}{A(z)}, \quad H(z) = \frac{1}{A(z)} \tag{8.4}$$

where $A(z)$ and $B(z)$ are FIR filters. The "AR" refers to the autoregressive part $a[n] * y[n]$ and the "X" to the extra input $b[n] * x[n]$. Although assuming that both the input and disturbance affect the output through systems with identical poles may seem physically unnatural, one big advantage of the ARX model is that the predictor defines a linear regression. We can add complexity to the ARX model by allowing for more freedom in describing the effect of the disturbance. For an additional FIR filter $C(z)$, we can use

$$G(z)=\frac{B(z)}{A(z)},\quad H(z)=\frac{C(z)}{A(z)} \tag{8.5}$$

This is called the ARMAX model due to the moving average component $c[n]*e[n]$.

The most general model we will consider here is the Box-Jenkins model, which provides for even more freedom in modeling of the disturbance effect. The Box-Jenkins model is given by

$$G(z)=\frac{B(z)}{F(z)},\quad H(z)=\frac{C(z)}{D(z)} \tag{8.6}$$

for FIR filters $F(z)$ and $D(z)$. Since room acoustics responses are well characterized by FIR filters, we intend to treat $G(z)$ as an FIR filter, and therefore the ARMAX and Box-Jenkins models will be sufficient to work with, and our choice between the two will depend on how we wish to model the disturbance effects.

Once we have chosen a model to work with, the next step is to estimate the parameters of the model, which are the filter tap values. We define the prediction error of a certain model with a vector of parameters $\boldsymbol{\theta}$ as

$$\varepsilon[n]=y[n]-\hat{y}[n|\boldsymbol{\theta}] \tag{8.7}$$

where $\hat{y}[n|\boldsymbol{\theta}]$ is an estimate of $y[n]$ given the past input and output data and a model with parameters $\boldsymbol{\theta}$. We can derive this estimate for the MMSE criterion[41]:

$$\hat{y}[n|\boldsymbol{\theta}]=h^{-1}[n,\boldsymbol{\theta}]*g[n,\boldsymbol{\theta}]*x[n]+(\delta[n]-h^{-1}[n,\boldsymbol{\theta}])*y[n] \tag{8.8}$$

Intuitively, then, a "good" model will be one with a small prediction error since it can describe the observed data well. We consider this prediction error sequence in (8.7) as a vector $\boldsymbol{\varepsilon}\in\Re^N$ and define the following norm:

$$V_N(\boldsymbol{\theta},\mathbf{Z}^N)=\frac{1}{N}\sum_{n=1}^{N}l(\varepsilon_n) \tag{8.9}$$

where $$\mathbf{Z}^N=\begin{bmatrix}y[1] & x[1] & y[2] & x[2] & \ldots & y[N] & x[N]\end{bmatrix} \tag{8.10}$$

is the input/output data set and $l(\varepsilon)$ is a scalar function. Now our parameter set estimate is given by the minimization of this norm:

$$\hat{\boldsymbol{\theta}}_N(\mathbf{Z}^N)=\arg\min_{\boldsymbol{\theta}}V_N(\boldsymbol{\theta},\mathbf{Z}^N) \tag{8.11}$$

A standard choice for $l(\varepsilon)$ is the quadratic norm, which is convenient for both computation and analysis and is given by

$$l(\varepsilon)=\tfrac{1}{2}\varepsilon^{2} \tag{8.12}$$

For the ARX model, say for (8.4) that

$$A(z)=1+\sum_{k=1}^{n_a} a_k z^{-k} \tag{8.13}$$

and

$$B(z)=1+\sum_{k=1}^{n_b} b_k z^{-k} \tag{8.14}$$

so

$$\theta=[a_1 \quad \ldots \quad a_{n_a} \quad b_1 \quad \ldots \quad b_{n_b}]^T \tag{8.15}$$

Now using the ARX model and the quadratic norm in (8.12), we find our one-step prediction estimate to be

$$\hat{y}[n|\theta]=\varphi_n^T\theta+\mu[n] \tag{8.16}$$

with a corresponding norm of

$$V_N(\theta,\mathbf{Z}^N)=\frac{1}{N}\sum_{n=1}^{N}\tfrac{1}{2}(y[n]-\varphi_n^T\theta-\mu[n])^2 \tag{8.17}$$

where

$$\varphi_n=\left[-y[n-1] \quad \ldots \quad -y[n-n_a] \quad x[n-1] \quad \ldots \quad x[n-n_b]\right]^T \tag{8.18}$$

and $\mu[n]$ is a known, data dependent vector. Finally, the least-squares estimate (LSE) can be obtained from minimizing (8.17) and is given by

$$\begin{aligned}\hat{\theta}_N^{LS}&=\arg\min_{\theta} V_N(\theta,\mathbf{Z}^N)\\ &=\left[\frac{1}{N}\sum_{n=1}^{N}\varphi_n\varphi_n^T\right]^{-1}\frac{1}{N}\sum_{n=1}^{N}\varphi_n(y[n]-\mu[n])\end{aligned} \tag{8.19}$$

Suppose the output was actually generated as

$$y[n]=\varphi_n^T\theta_0+\mu_0[n] \tag{8.20}$$

and let

$$\mathbf{R}_\infty=E[\varphi_n\varphi_n^T] \tag{8.21}$$

and

$$\mathbf{h}_\infty=E[\varphi_n\mu_0[n]] \tag{8.22}$$

as $N\rightarrow\infty$. It can be shown that for the LSE in (8.19) to be a consistent estimator, we require that $\mathbf{R}_\infty$ is nonsingular and that $\mathbf{h}_\infty=\mathbf{0}$. This is typically the case when $x[n]$ and $\mu_0[n]$ are independent, the covariance matrix of the input sequence is nonsingular, and $\mu_0[n]$ is white noise. Also, we can apply the central limit theorem[17] to show that the distribution of

$$\sqrt{N}\left(\hat{\theta}_N^{LS}-\theta_0\right) \tag{8.23}$$

converges to the normal distribution with zero mean and covariance $\mathrm{var}\{\mu_0[n]\}\mathbf{R}_\infty^{-1}$.

A variety of non-parametric methods exist that allow for the estimation of the frequency response of a system. A simple method is to estimate

$$\hat{G}_N(\omega) = \frac{Y(\omega)}{X(\omega)} \tag{8.24}$$

where $X(\omega)$ and $Y(\omega)$ are the N-point Fourier transforms of $x[n]$ and $y[n]$ respectively. This is called the empirical transfer-function estimate (ETFE). It can be shown that if the input is periodic, then

- the ETFE is defined only for a fixed number of frequencies,
- the ETFE is unbiased,
- and the variance of the ETFE decays like $1/N$ where N is a multiple of the period.

If the input is the realization of a stochastic process, then

- the ETFE is an asymptotically unbiased estimate of the transfer function,
- the variance of the ETFE does not decrease as N increases and is instead given by the noise-to-signal ratio,
- and the ETFE estimates at different frequencies are asymptotically uncorrelated.[41]

In order to concoct an estimator that decreases in variance as the data size increases, we can smooth the ETFE. One such smoothing technique is the Blackman-Tukey procedure, so that

$$\hat{G}_N(\omega) = \frac{\hat{\Phi}_{yx}(\omega)}{\hat{\Phi}_{xx}(\omega)} \tag{8.25}$$

where $\hat{\Phi}_{yx}(\omega)$ and $\hat{\Phi}_{xx}(\omega)$ are estimates of the cross-spectrum between the input and output and the auto-spectrum of the input respectively. We implement these estimates using weighted periodograms:

$$\hat{\Phi}_{xx}(\omega) = \int_{-\pi}^{\pi} W_\gamma(\xi - \omega) | X(\xi) |^2 \, d\xi \tag{8.26}$$

$$\hat{\Phi}_{yx}(\omega) = \int_{-\pi}^{\pi} W_\gamma(\xi - \omega) Y(\xi) X^*(\xi) d\xi \tag{8.27}$$

for some weighting frequency window $W_\gamma(\omega)$.

8.3 Experimental Results

Our approach toward analyzing the data collected in the manner described in Section 8.1 is to apply the system identification tools developed in Section 8.2 to understand the properties of the systems $G_{12}(z)$ and $G_{21}(z)$. We can use the ARX, ARMAX, or Box-Jenkins models to obtain an initial guess during stationary periods for adaptive algorithms such as the decorrelation approach and the EM algorithm, as discussed in Chapter 7. One drawback is that the room impulse responses tend to be long, requiring hundreds of samples to accurately represent the signal. The decorrelation method developed in [35] becomes numerically unstable for large system orders, so this method is undesirable for the application. The results here are presented for the sentence "Why were you away a year Roy?" with a male subject and car noise played in the background (approximately 12 dB below the speech level at the microphones). The microphones are not perfectly matched, and so any difference in frequency response will be lumped in with the unknown systems to be estimated.

The subjects each read a series of 12 sentences in the noisy environment and without noise. By analyzing the pauses between each of these sentences, we can get a good feeling for the stationarity of the system.

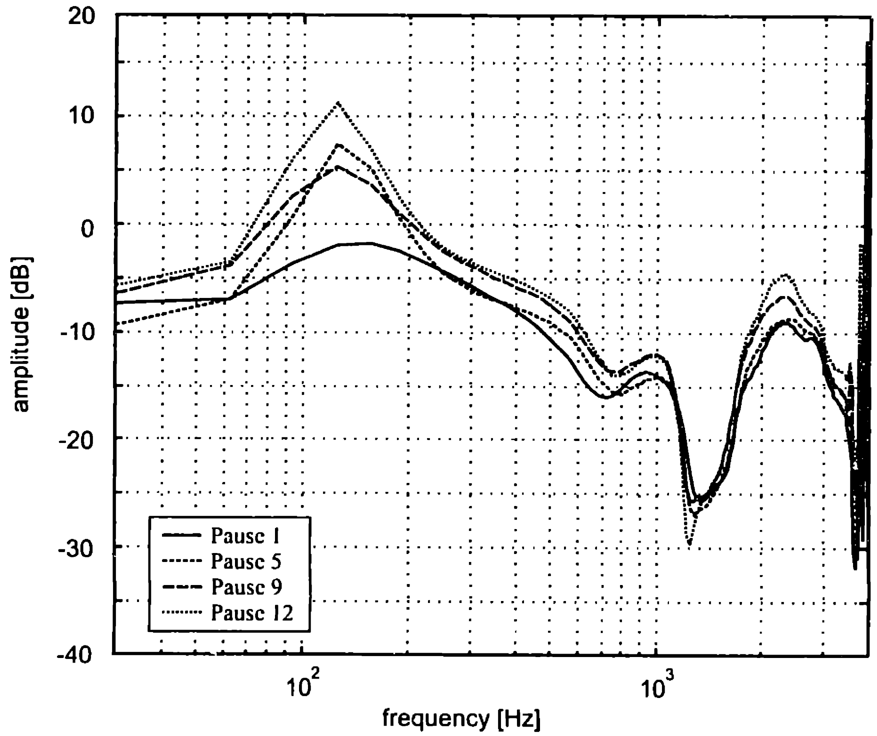


Figure 8.4 The magnitude of estimates of $G_{21}(\omega)$ during pauses between sentences, computed using the Blackman-Tukey procedure.

We expect that $G_{21}(z)$ will be a slowly time-varying system due to the possible physical movement of the handset. Spectral estimates based on the Blackman-Tukey procedure are given in Figure 8.4 and Figure 8.5 for four different pauses.

The phase is quite stationary over the testing period, while there is only a significant discrepancy at low frequencies for the amplitude. The high frequency discrepancy is due to an IRS weighting filter that was applied to the original clean speech, which causes an increase in the variance of the spectral estimates.

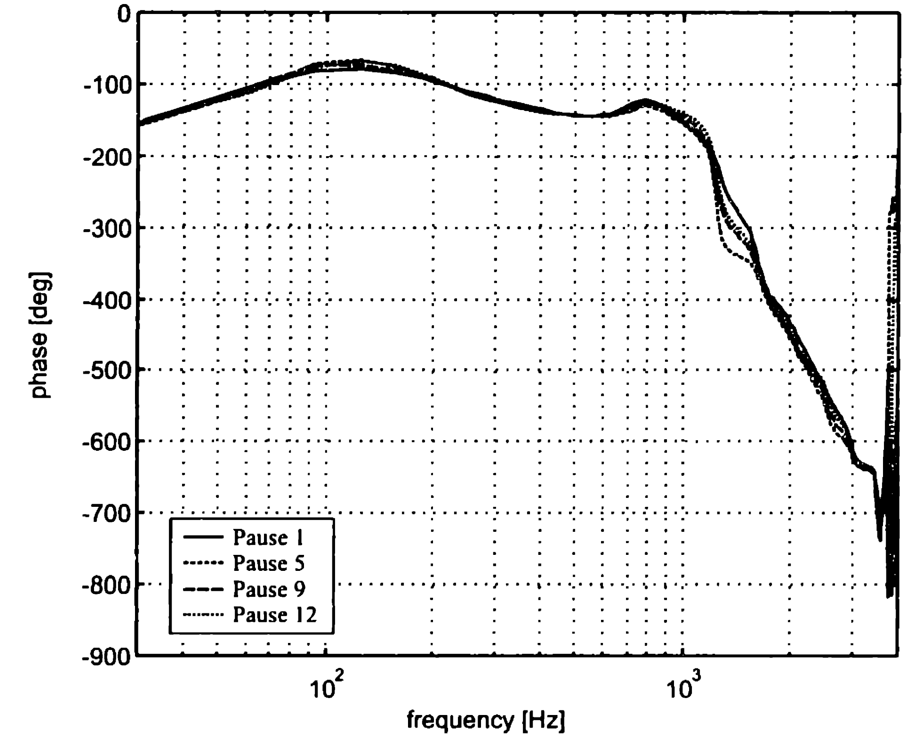


Figure 8.5 The phase of estimates of $G_{21}(\omega)$ during pauses between sentences, computed using the Blackman-Tukey procedure.

We can oerform the same analysis to estimate $G_{12}(\omega)$, as shown in Figure 8.6 and Figure 8.7 by using the noise-free recording of the 12 sentences. There is a disturbing effect around 1.3 kHz where a zero is found quite close to the unit circle. This could be a result of a nonminimum phase acoustic transfer function or from a problem with the experiment setup. Whatever the cause, this spike causes stability problems during the second stage of the reconstruction. We can use the Nyquist stability criterion to see this effect. For the second stage of reconstruction to be causal and stable, we require that no zeros of $1 - G_{12}(z)G_{21}(z)$ be outside the unit circle. This is equivalent to having no zeros of $1 - G_{12}(z^{-1})G_{21}(z^{-1})$ inside the unit circle.

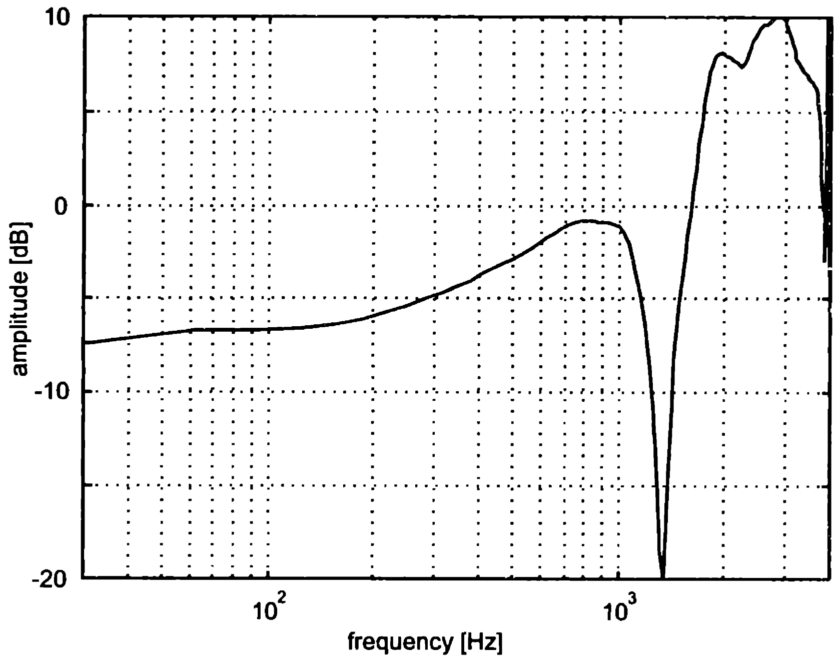


Figure 8.6 The magnitude of an estimate of $G_{12}(\omega)$ during a noise-free recording of the subject, computed using the Blackman-Tukey procedure.

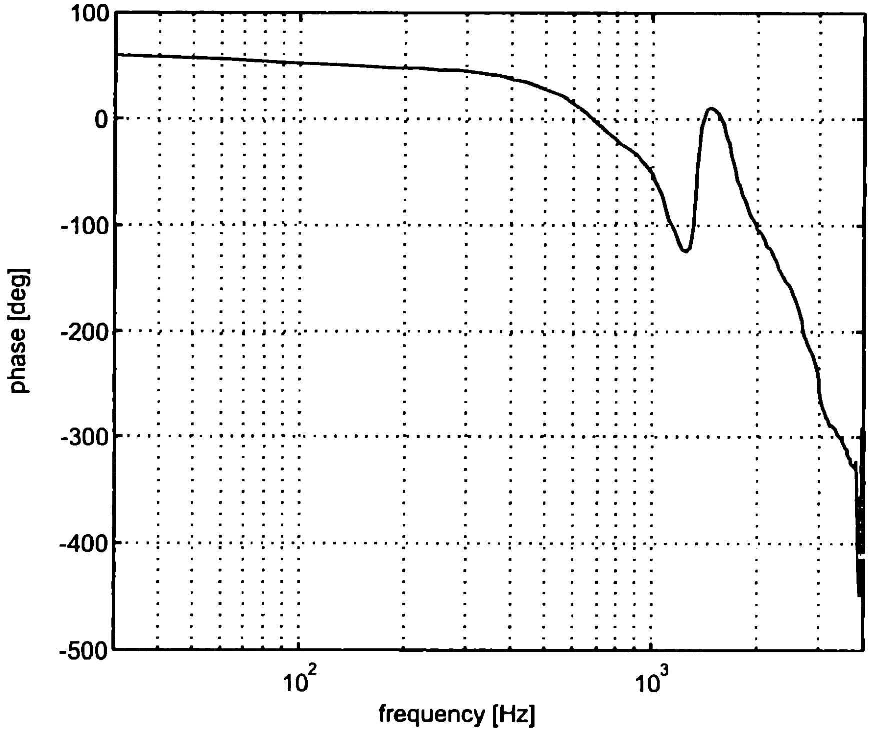


Figure 8.7 The phase of an estimate of $G_{12}(\omega)$ during a noise-free recording of the subject, computed using the Blackman-Tukey procedure.

By the Encirclement Property[42], the Nyquist plot of $G_{12}(\omega)G_{21}(\omega)$ must not encircle 1 in a clockwise direction for the system to be stable. Such a Nyquist plot is shown around 1 in Figure 8.8. Clearly, the Nyquist stability criterion is not met, and the 1 is encircled in a clockwise direction at around 2.4 kHz.

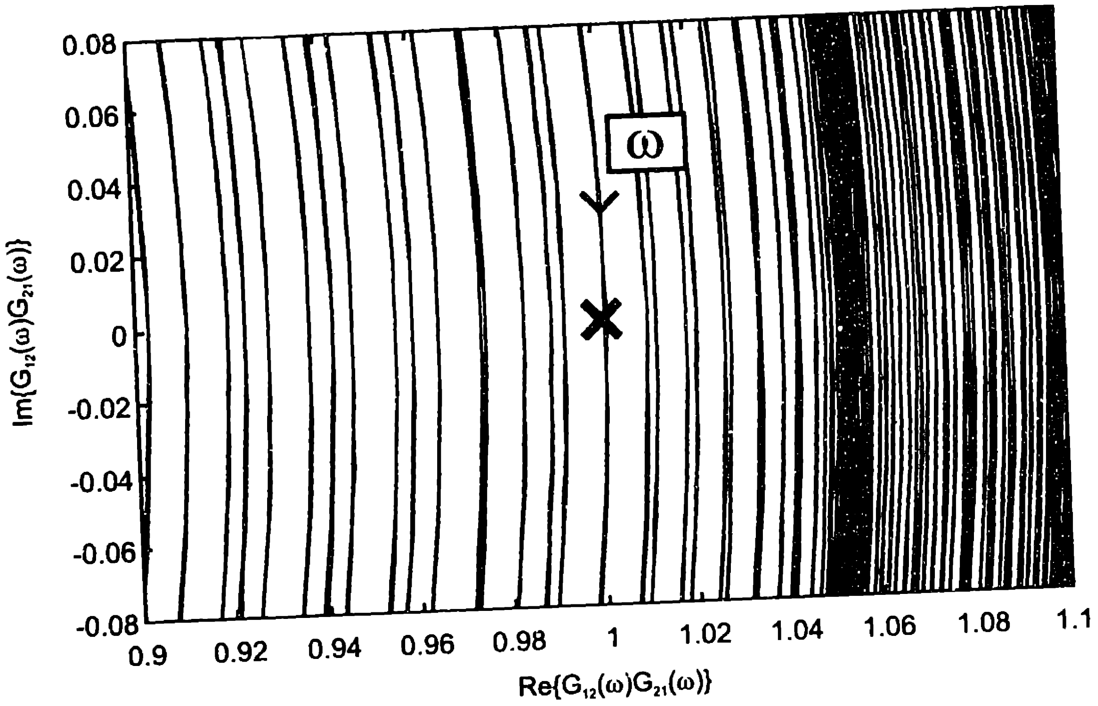


Figure 8.8 Nyquist plot of $G_{12}(\omega)G_{21}(\omega)$ in the complex plane. The point (1,0) is marked by the "X", and the direction of increasing ω is indicated by the arrow.

Due to this stability problem, attempting to apply the second stage of the reconstruction filter causes annoying reverberation effects that worsen the quality of the enhanced speech. Since there is a much lower level of clean speech in the secondary microphone than in the primary microphone, the reverberation effects that the second stage of the reconstruction is attempting to correct for might not be terribly annoying or perceptible. In addition, the second stage does not reduce the noise energy much in general, so an adequate prediction of the system's performance can be observed after only the first reconstruction stage. Using a fixed (non-adaptive) reconstruction filter, we noticed little signal distortion but much less noise reduction than what some of the single microphone techniques are capable of. There was some noticeable noise distortion, but it is believed that this was mainly due to the estimation problems at high frequencies caused by the IRS weighting filter. The failure of this technique to achieve the high distortion-free noise suppression that we witness with single microphone techniques is attributed to the inaccuracies in estimating the systems $G_{12}(\omega)$ and $G_{21}(\omega)$. This is partly due to the non-adaptive nature of the techniques and also to possible minimum phase components in the acoustic transfer functions.

Although the decorrelation approach developed in [35] fails to converge for high FIR filter orders, we were able to obtain slow convergence for a 10th order model. The adaptation of each of the 11 filter coefficients for each transfer function is shown in Figure 8.9 and Figure 8.10 for the same noisy sentence used above.

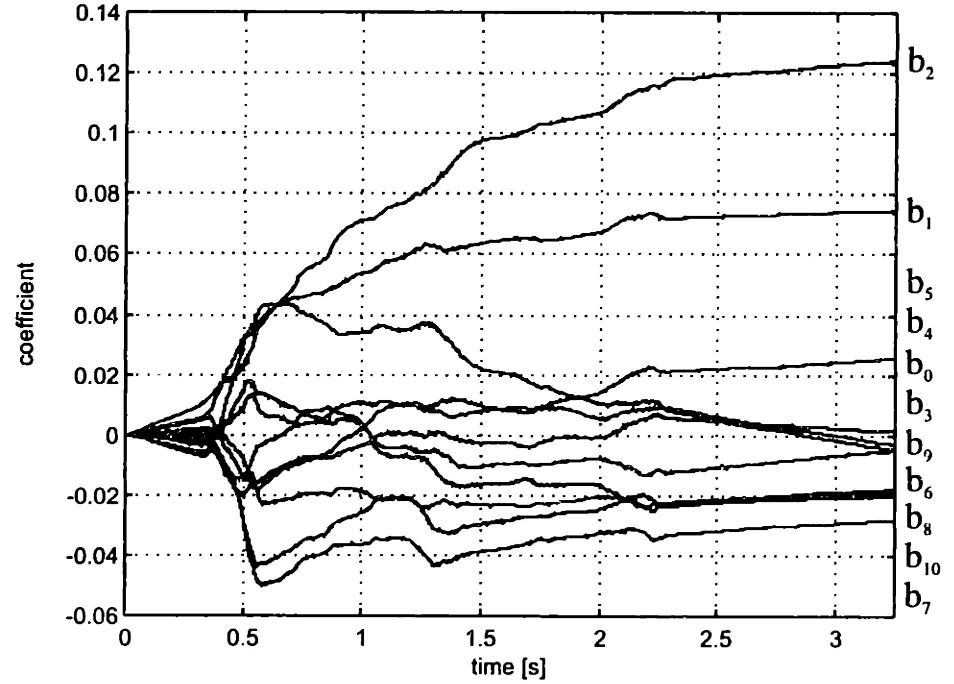


Figure 8.9 The adaptation of the coefficients of $H_{21} = \sum_{k=0}^{10} b_k z^{-k}$ using the decorrelation algorithm presented in Section 7.2.

The coefficients were all initialized to zero. When the speech begins just before half a second into the experiment, the coefficients spread out quickly and then gradually settle around their final values. The frequency responses of these filters at the end of the sentence are shown in Figure 8.11 and Figure 8.12. The estimates are rather poor. The noise reduction is marginal, and both signal and noise distortion are evident but not terribly annoying.

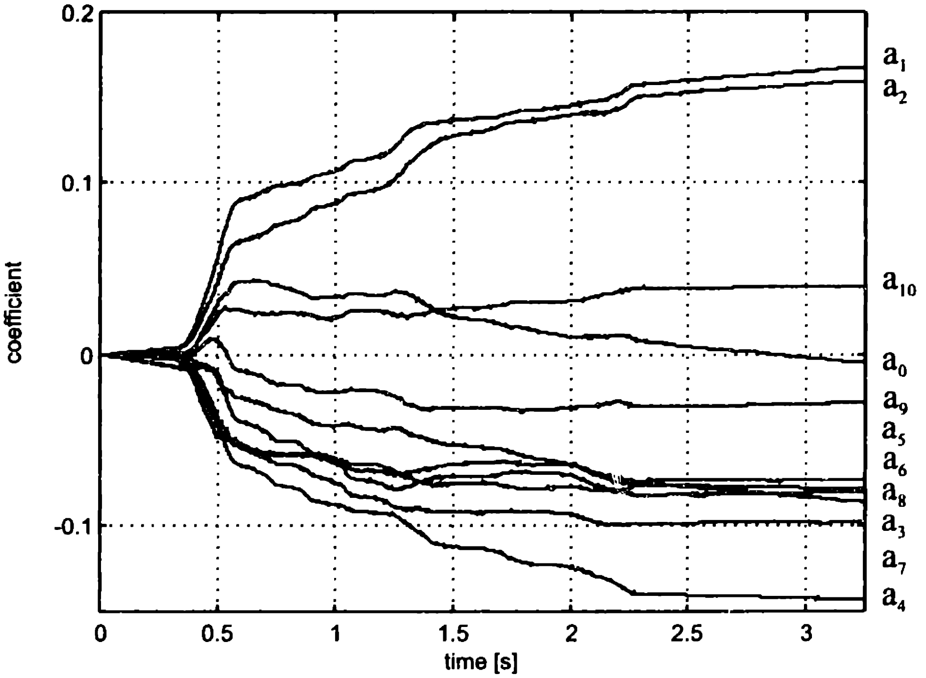


Figure 8.10 The adaptation of the coefficients of $H_{12} = \sum_{k=0}^{10} a_k z^{-k}$ using the decorrelation algorithm presented in Section 7.2.

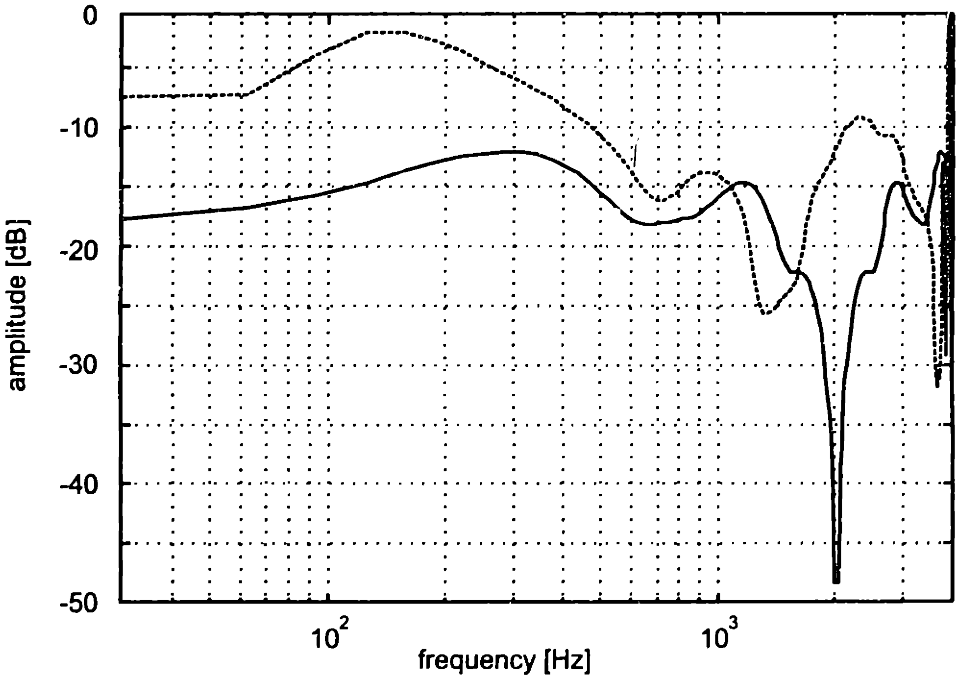


Figure 8.11 The magnitude of an estimate of $G_{21}(\omega)$ from the 10th order decorrelation algorithm *(solid)*, compared with the estimate from Figure 8.4 *(dashed)*.

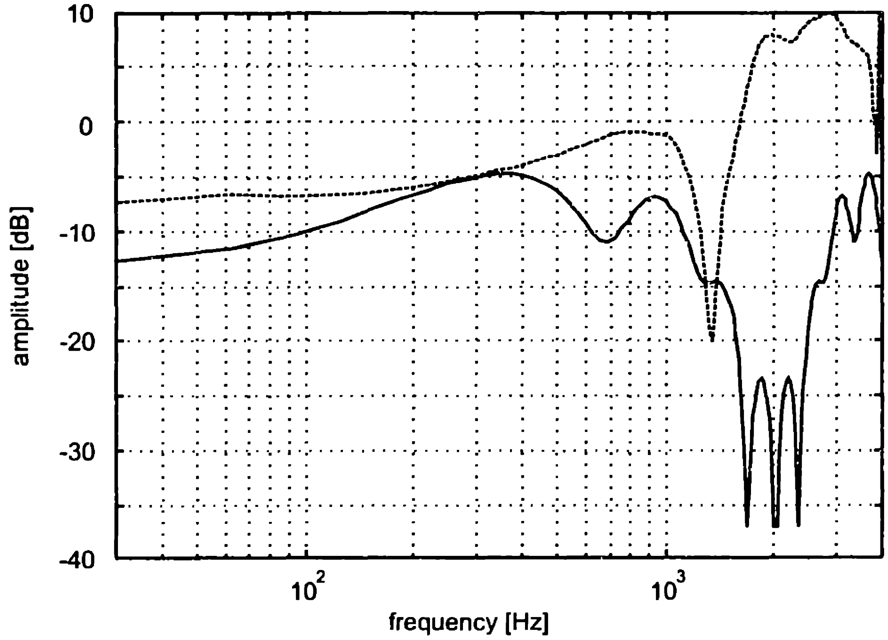


Figure 8.12 The magnitude of an estimate of $G_{12}(\omega)$ from the 10^{th} order decorrelation algorithm *(solid)*, compared with the estimate from Figure 8.6 *(dashed)*.

9

A Stationary Simplification

- *Experimental Setup and Restrictions*
- *Software Operation*
- *Results and Error Analysis*

The problems encountered in Chapter 8 are difficult to ıesolve given the inflexibility of the experimental setup. Here we attempt to simplify the problem and provide tighter control over the various factors that could cause trouble.

9.1 Experimental Setup and Restrictions

The main difficulty encountered in Chapter 8 was the uncertainty of the room acoustic transfer functions. This made it difficult to determine the frequency responses of the systems and their invertibility. Our new test setup takes place in a sound booth, which is shown in Figure 9.1. This booth is effectively shielded from all outside acoustic disturbances, but is not anechoic.

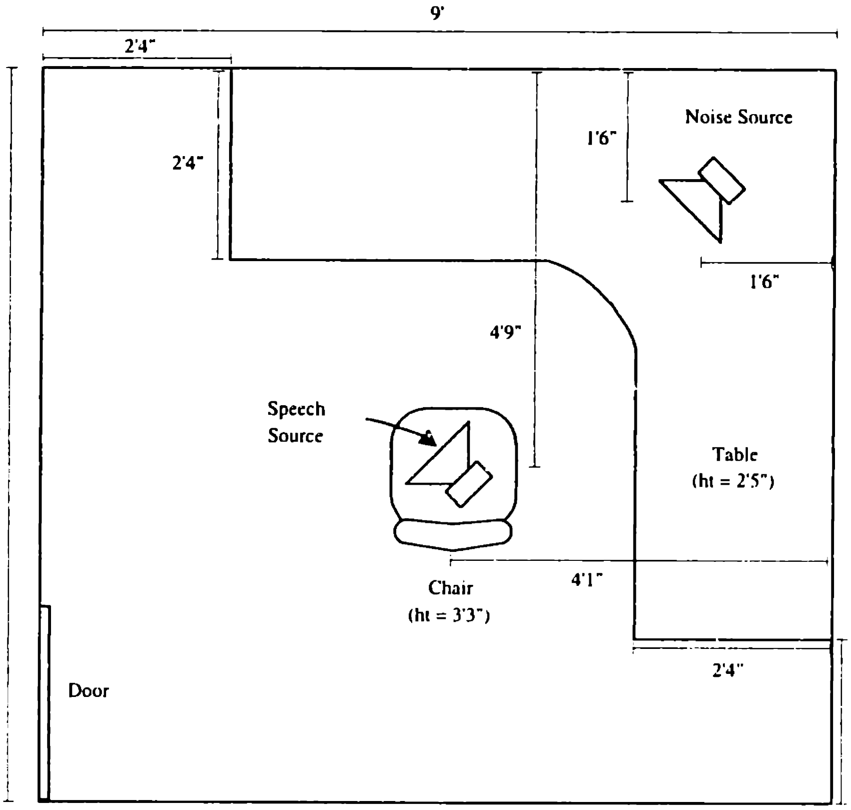


Figure 9.1 Experimental setup in a sound booth for 2 sound sources and 2 microphones. The microphones are affixed to the speech source, as shown in Figure 9.2.

There are two sound sources. The noise source is a single speaker positioned in the corner of the booth and is elevated on a table, pointed towards the center of the room. The clean speech source is stationed on a chair near the middle of the room. A cell phone is outfitted with two Carlson tube microphones and fastened to the clean speech loudspeaker as shown in Figure 9.2. The rectangular shape of the loudspeaker and the position of the cell phone were chosen in order to approximate head diffraction effects that occur when a person uses a cell phone.[43] Also, we are justified in assuming stationary acoustic systems in this environment since all source and microphone positions are fixed.

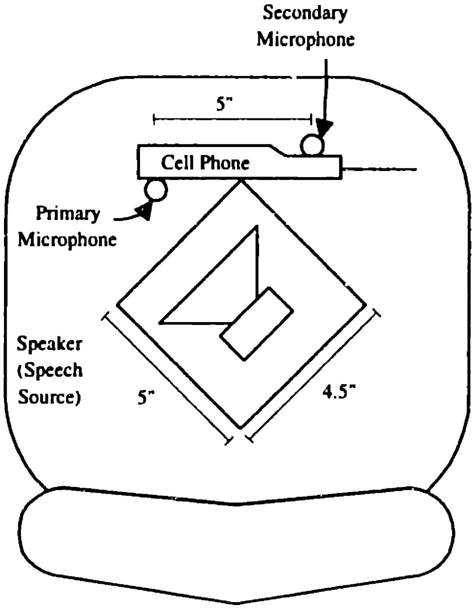
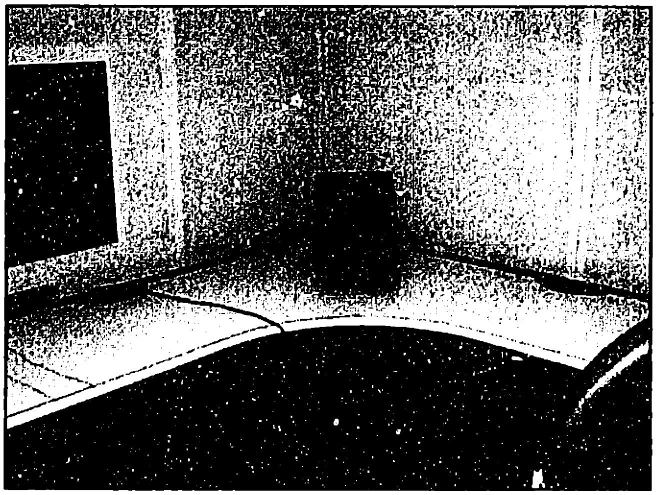


Figure 9.2 Close-up of the chair in Figure 9.1, showing the speech source and the cell phone outfitted with both microphones.

Figure 9.3 through Figuıe 9.6 are photos of the actual experimental зetup



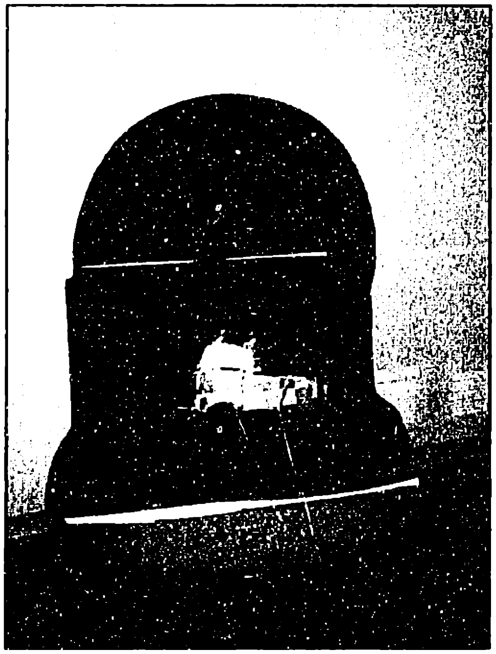


Figure 9.3 View of the noise source from the door of the sound booth, behind the speech source.

Figure 9.4 The chair containing the speech source and microphones.

Note the use of "Sound Soak" foam around the speech source that absorbs sound, preventing immediate reflections from the chair.

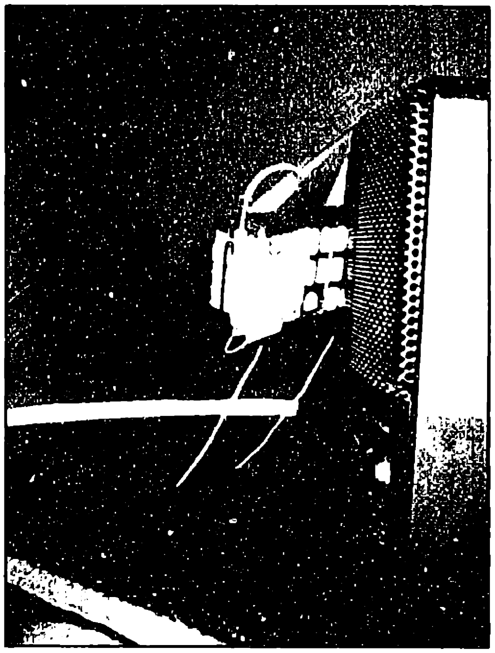


Figure 9.5 Position of the primary microphone with respect to the speech source.

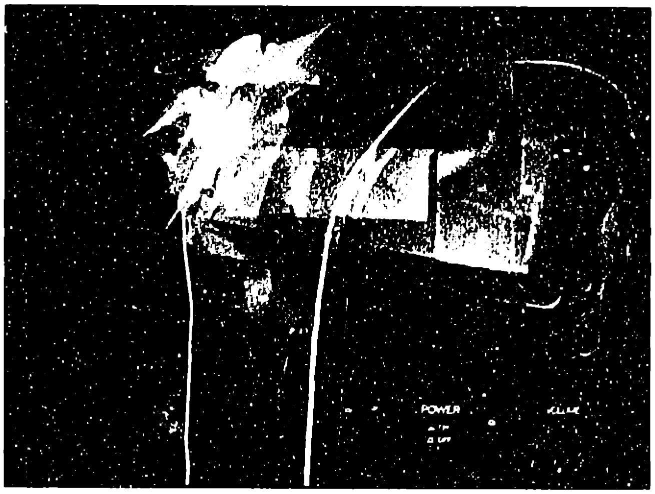


Figure 9.6 Close-up of the cell phone. The secondary microphone and body of the primary microphone are visible.

Wires connecting the microphones and loudspeakers are passed through a hole in the sound booth to an outside control station. The electronics setup is illustrated in Figure 9.7. The two loudspeakers used have built-in amplifiers, and the Carlson tube microphones interface with customized receivers. These microphone signals are subsequently amplified with instrumentation amplifiers before leaving the sound booth. Both a PC running Linux and a pair of DAT players are connected to the system as shown. In this manner, the noise sources can be driven with sound from the PC or recordings on the DAT. In addition, the microphone outputs can be either recorded on a DAT or returned to the PC. This provides the flexibility for both real-time and offline analysis of the system. The system identification software running on the PC (see Section 9.2) can have access to both the speaker inputs and microphone outputs, facilitating the identification of the relevant acoustic systems. Additionally, pre-recorded noise can be used to drive the speakers while the output at the microphones is recorded. The results can be analyzed at a later time using the techniques from Chapter 8.

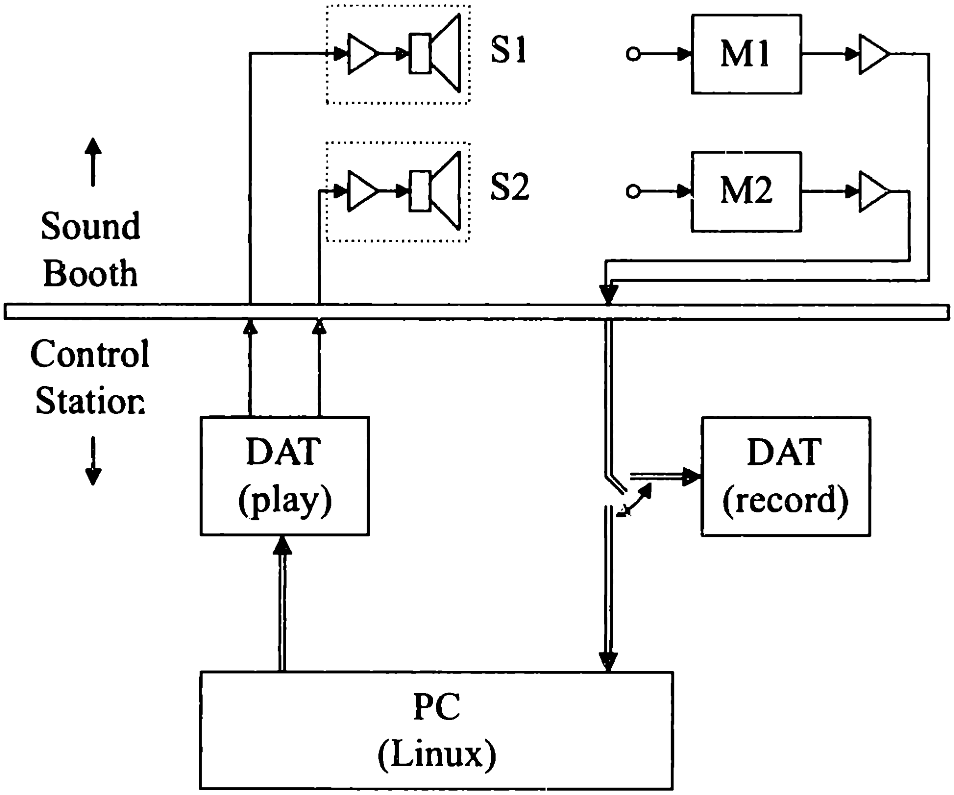


Figure 9.7 Electronics setup for the sound booth experiment. The microphones M1, M2 and the speaker sound sources S1, S2 interface with a pair of DAT players and a PC.

9.2 Software Operation

The PC uses a MultiSound Pinnacle card from Turtle Beach to interface with the remainder of the system. System identification software (a clone of SYSid from SYSid Labs that runs under Linux) makes use of this card and allows for the accurate estimation of the relevant acoustic systems.

As we have control over the input sequence when attempting to identify an LTI system, the problem of system identification becomes much simplified. Consider using a periodic input sequence with period R and total length N and grouping the output sequence into M batches, so that $N = R \cdot M$. We are careful to throw away the first batch of output samples, since transient effects can complicate our analysis. We then average the output sequence in the remaining groups, and divide the spectrum of this averaged signal with that of one period of the input to obtain an estimate of the transfer function of the system.

This is a means of smoothing the ETFE (recall Section 8.2). Note that another way to smooth the ETFE is to split the data set into M batches each containing $R = N/M$ samples and then to average the ETFEs corresponding with each batch. Say $\hat{G}_R^{(k)}(\omega)$ is the ETFE for batch k and $\hat{G}_N(\omega)$ is the average ETFE. Then we have

$$\hat{G}_N(\omega) = \frac{1}{M}\sum_{k=1}^{M}\hat{G}_R^{(k)}(\omega) \tag{9.1}$$

Since these data sets do not overlap, the corresponding ETFEs are independent and

$$\mathrm{var}\{\hat{G}_N(\omega)\} = \frac{1}{M}\mathrm{var}\{\hat{G}_R^{(k)}(\omega)\} \tag{9.2}$$

Therefore, the variance of our estimator drops by $1/M$. Also, from Section 8.2 we know that the variance of an ETFE of a periodic signal drops by $1/N$ where N is a multiple of the period. This implies that we could alternately implement our estimator by taking a DFT over the entire output signal instead of averaging the individual groups.

The idea is to take a large enough input signal to guarantee that the variance of our estimate is below our error tolerance. We use a chirp signal for the input so that energy is distributed across all frequencies of interest for each period. Figure 9.8 demonstrates the performance of this setup.

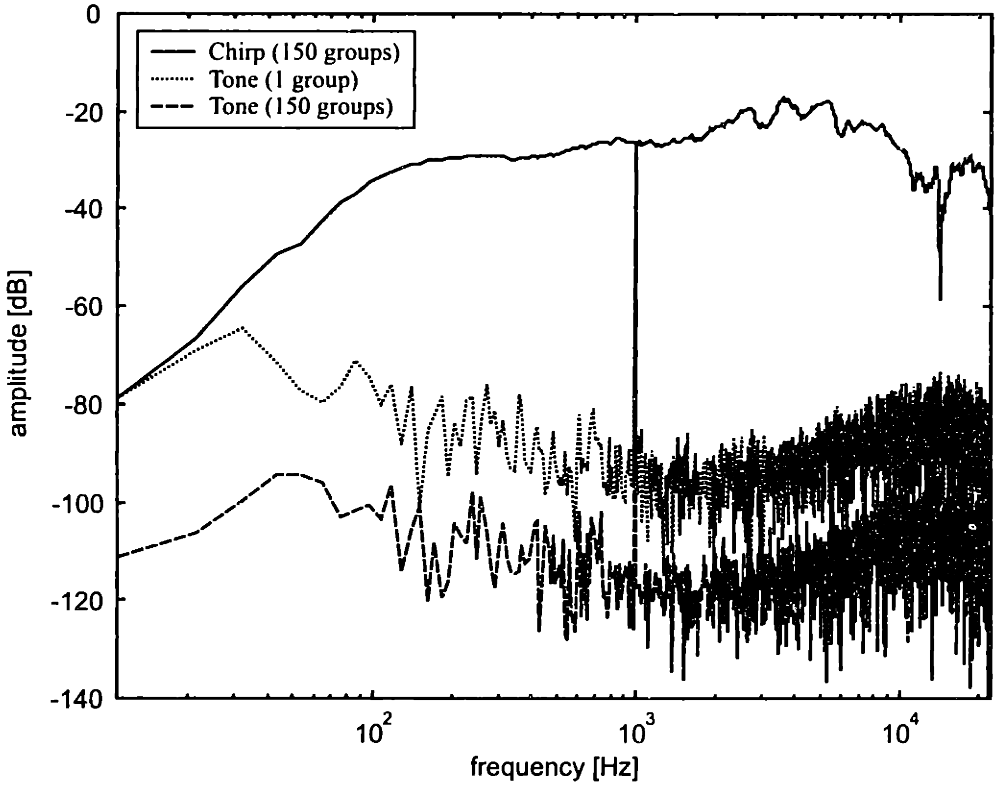


Figure 9.8 Reducing the noise floor and spectral leakage through averaging.

In Figure 9.8, the frequency response of an acoustic system was first estimated using a chirp signal as an input. Here we use a 4096-sample FIR filter model, and set $M = 150$. We then use a

1 kHz tone as an input to verify the accuracy of our estimate of the transfer function at this frequency. The result of using this tone with $M = 150$ and $M = 1$ is shown so the spectral leakage can be compared. We see that both estimates from the tones agree with the magnitude of the spectral estimate at 1 kHz, although with $M = 150$ the noise floor is reduced by approximately 30 dB. We can reduce this noise floor to an arbitrarily low level by increasing the number of averages performed.

9.3 Results and Error Analysis

Using the technique outlined above, we calculated impulse responses 65,536 samples long with $M = 150$ for four acoustic systems in the sound booth. The results are given in Figure 9.10. We see that the speech to primary microphone impulse response, $h_{11}[n]$, is very short and approximately a delta function. There are no reflections for this short path. The path from the speech to the secondary microphone, $h_{12}[n]$, is not as clean and there is a single reflection evident, presumably off the wall or ceiling of the sound booth. The systems from the noise source to the microphones are more distorted and contain more reflections.

The different energies of the impulse responses are mainly due to microphone and amplification mismatches.

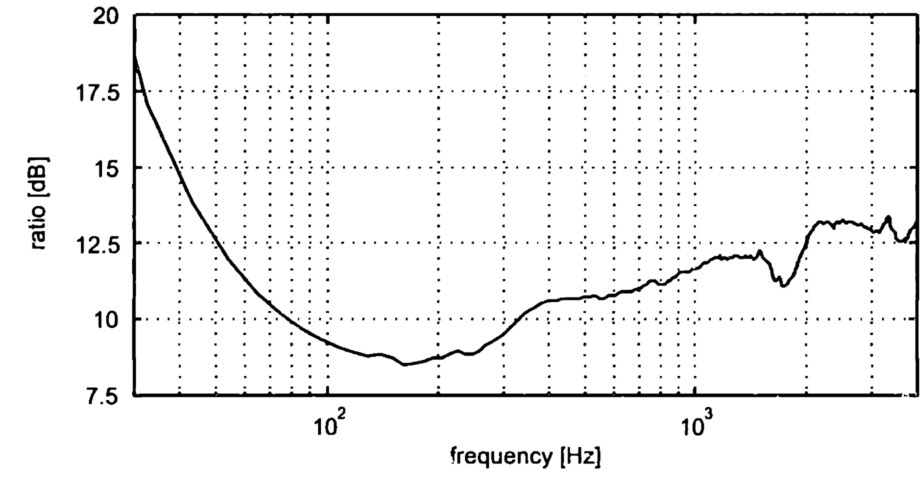


Figure 9.9 Ratio of the magnitude of the frequency response of the secondary microphone to that of the primary microphone.

To correct for this, we measure the impulse responses from a speaker to each of the two microphones placed nearby each other, all in an approximate free-field environment. After dividing the magnitude of the frequency response of the secondary microphone with that of the primary, we obtain the curve given in Figure 9.9. Before further processing, we correct for this microphone mismatch by scaling the response of the primary microphone.

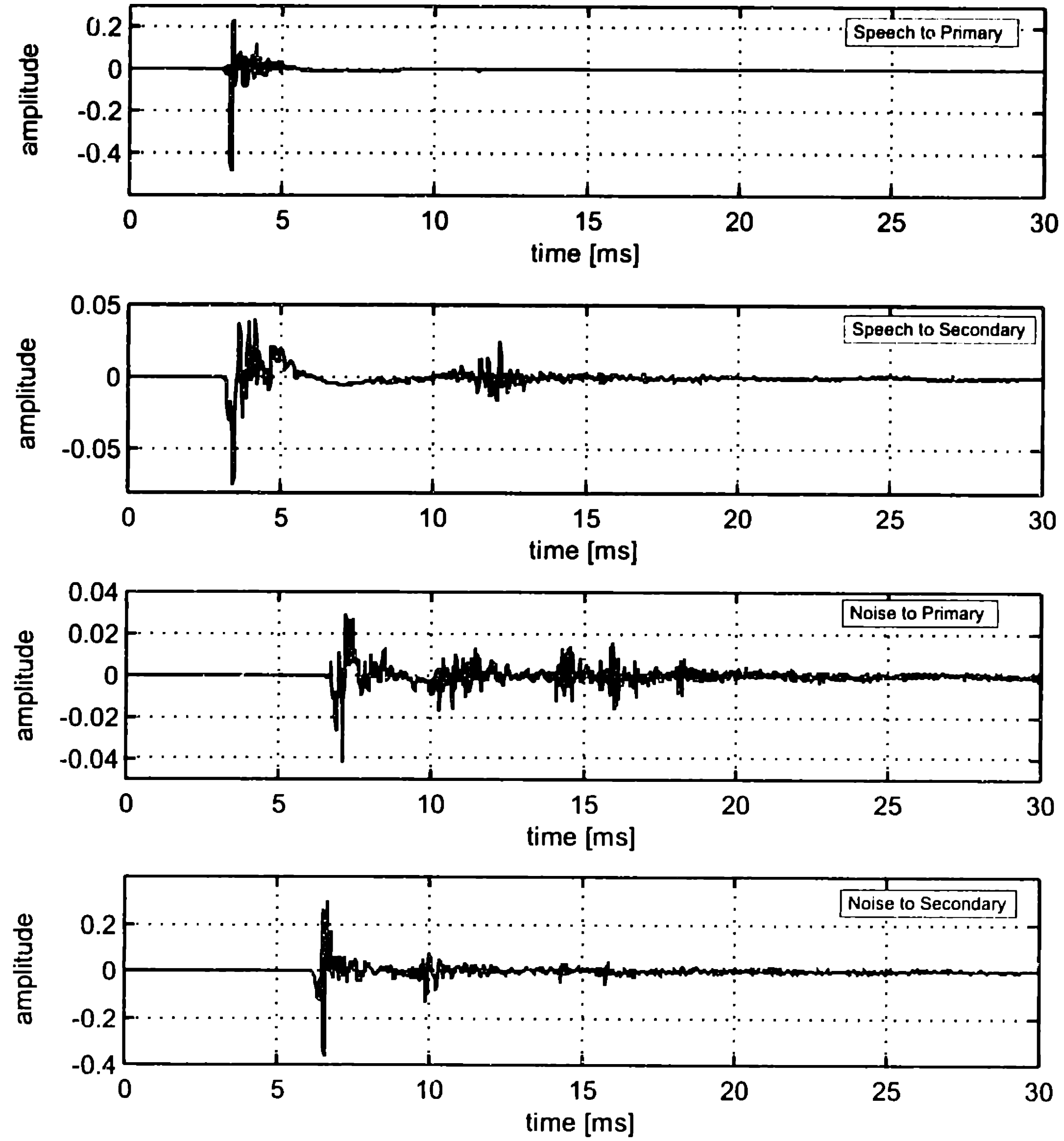


Figure 9.10 Impulse responses for the relevant acoustic systems estimated using 65,536-point DFTs and 150 averages.

Our next step is to determine whether $H_{11}(z)$ and $H_{22}(z)$ have causal and stable inverses after a finite delay. To do this, we will factor each system into a minimum phase component and an all-pass component:

$$H(z) = H_{min}(z)H_{ap}(z) \tag{9.3}$$

We take advantage of the fact that the complex cepstrum of a system is causal if and only if the system is minimum phase, and apply homomorphic filtering as described in [27] and shown in Figure 9.11. First we compute the real cepstrum $c_h[n]$. If $h[n]$ were minimum phase, we could then reconstruct the complex cepstrum from the real cepstrum by multiplying by

$$\ell_{min}[n] = 2u[n] - \delta[n] \tag{9.4}$$

However, for a more general $h[n]$, $\ell_{min}[n] \cdot c_h[n]$ will result in the complex cepstrum of the minimum phase sequence that has the same Fourier transform magnitude as $h[n]$, which is precisely the complex cepstrum of $h_{min}[n]$, denoted as $\hat{h}_{min}[n]$. We then divide by this minimum phase component to obtain the all-pass system

$$H_{ap}(\omega) = \frac{H(\omega)}{H_{min}(\omega)} \tag{9.5}$$

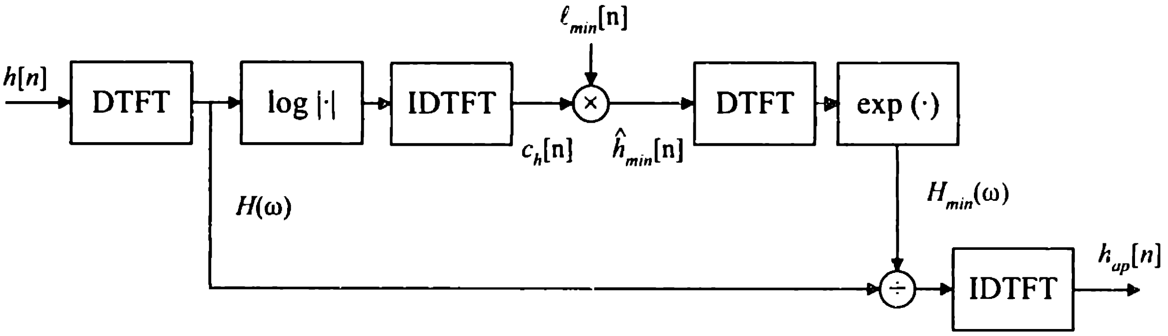


Figure 9.11 Block diagram of the computation of the all-pass component of a system by factoring out the minimum phase component using homomorphic filtering.

If the original system $h[n]$ were simply a minimum phase system with possibly some finite negative delay, we expect the all-pass factor $h_{ap}[n]$ to have linear phase. By analyzing the group delay of the all-pass system

$$-\frac{d}{d\omega}\arg[H_{ap}(\omega)] \tag{9.6}$$

we can determine whether the system is minimum phase and if not, and locate the frequencies where the nonminimum phase zeros occur. Such a group delay plot for the noise to secondary microphone system, $H_{22}(z)$, is shown in Figure 9.12. There is no doubt that this system is nonminimum phase, and has a number of nonminimum phase zeros indicated by the spikes in Figure 9.12. This nonminimum phase characteristic impacts the computation of

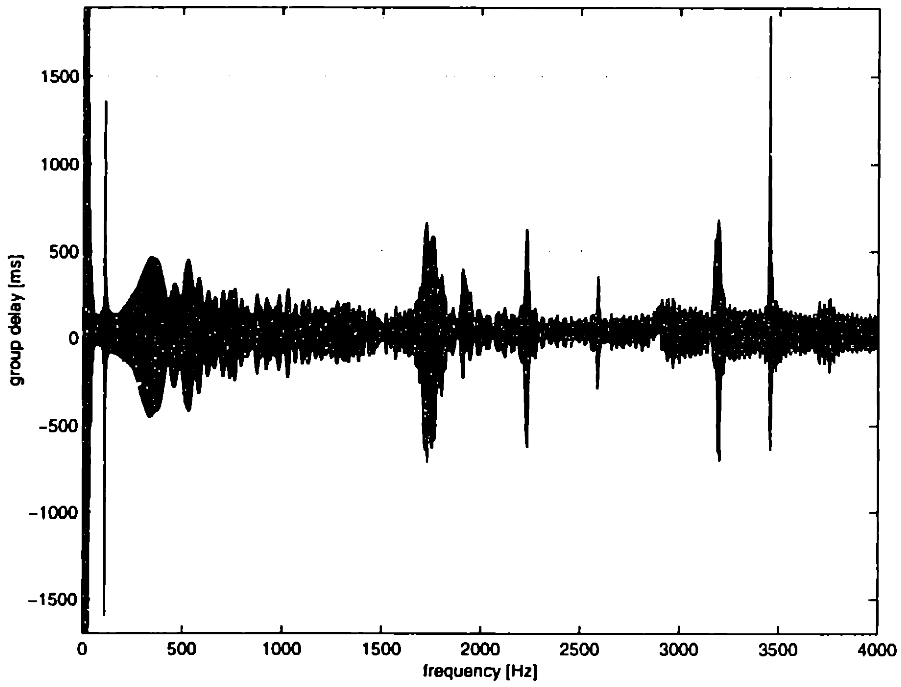


Figure 9.12 Group delay of the all-pass component of our estimate of $H_{22}(z)$.

$G_{21}(\omega) = H_{22}^{-1}(\omega)H_{21}(\omega)$, where the inverse of $H_{22}(\omega)$ is calculated by simply inverting the transfer function. Such an attempt at computing $G_{21}(\omega)$ is shown in Figure 9.13. The frequency regions of no interest can be filtered out, eliminating the stability problems at low frequencies. However, there are nonminimum phase zeros which we cannot address in this manner.

If the acoustic systems were nonminimum phase but only because of a pure finite delay, we could compensate by introducing delay into the reconstruction system. For example, consider $G_{21}(\omega) = H_{22}^{-1}(\omega)H_{21}(\omega)$. This system will have a negative delay if the noise reaches the primary microphone before the secondary microphone, as the delay for $H_{21}(\omega)$ will be less than that for $H_{22}(\omega)$. This will be expected whenever the noise source is closer to the mouthpiece than it is to the back of the cell phone. Therefore, negative system delays will be a common occurrence for the application and must be dealt with.

Suppose $G_{21}(z)$ is a causal system composed with a negative delay of T_1 samples, and $G_{12}(z)$ is a causal system composed with a negative delay of T_2 samples. We can use delayed versions of these systems:

$$\tilde{G}_{21}(z) = z^{-T_1}G_{21}(z) \quad (9.7)$$

$$\tilde{G}_{12}(z) = z^{-T_2}G_{12}(z) \quad (9.8)$$

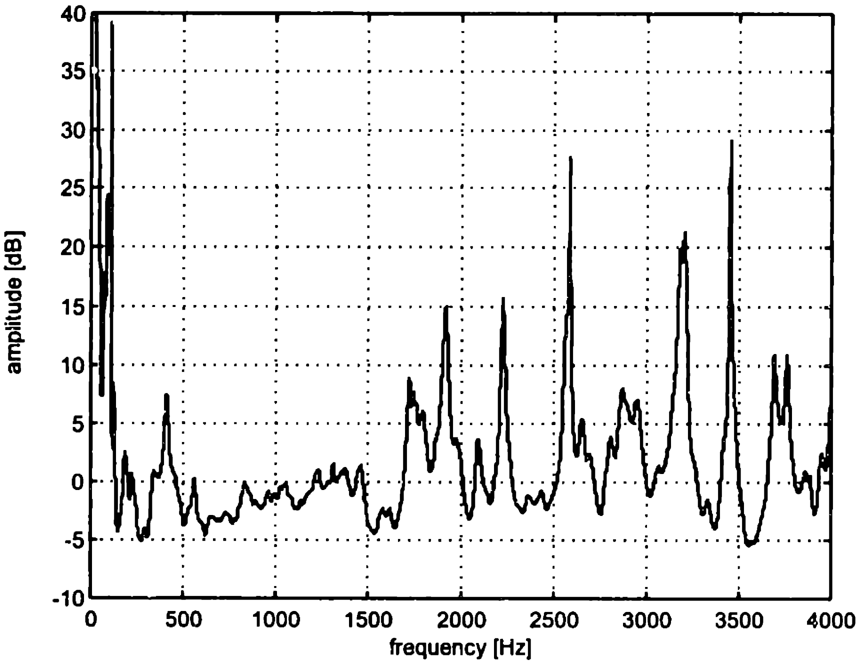


Figure 9.13 An attempt at computing $G_{21}(\omega) = H_{22}^{-1}(\omega)H_{21}(\omega)$ that demonstrates the problem of introducing tonal artifacts when $H_{22}(\omega)$ in nonminimum phase.

and realize a causal reconstruction system as given in Figure 9.14, where $s_d[n]$ and $w_d[n]$ are delayed versions of $s[n]$ and $w[n]$.

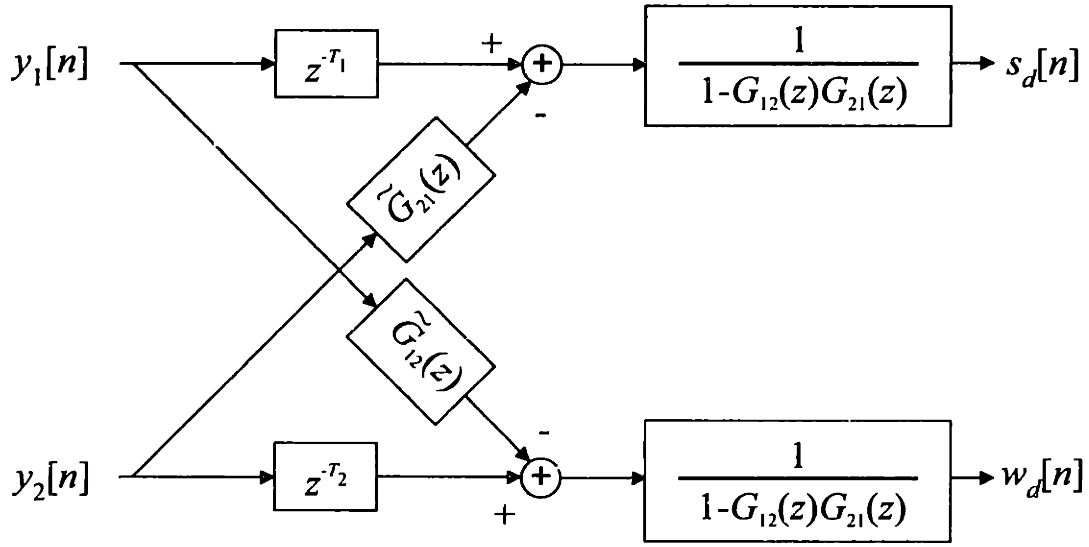


Figure 9.14 Reconstruction system modified to handle subsystems with negative delay.

This method results in a good deal of noise reduction, but the stability problems introduce tonal noise akin to the musical noise witnessed in single microphone speech enhancement that makes the enhanced speech less desirable than the original noisy speech.

10

Discussion

- *Summary of Results*
- *Future Work*

10.1 Summary of Results

Our first main accomplishment was to recognize the general form of Malah's modified MMSE-LSA speech enhancement scheme. Many modules in Malah's algorithm provide functionality desired by other speech enhancement applications. By using different core estimators within this framework, the overall system can be better tailored to meet application requirements. Due to the shared modules, several enhancement schemes of this form can be run concurrently with a small increase in computational complexity. This allows for the possibility of using different core estimators for different functions in the same application, as in our example of the pre-processor for the IS-641 speech coder.

We found that using a slightly modified version of Ephraim's signal subspace enhancement approach as a core estimator results in a very flexible algorithm that allows for the explicit specification of the tradeoff between signal distortion and noise reduction for each frequency bin. The resulting hybrid algorithm performed well when suppressing car noise. The spectral algorithm proved effective in suppressing babble, and a combination algorithm that used the spectral algorithm as a pre-processor for the LPC analysis of the IS-641 and the hybrid algorithm as a pre-processor for the remainder of the coder demonstrated good robustness properties with respect to noise type.

One of the biggest difficulties faced when preparing subjective listening tests was to determine the aggression level used for each algorithm. When noise is not overwhelmingly loud, the noisy speech might be preferred to an enhanced version with a reduced noise level but an unnatural sound. A perceptual model of the tradeoffs between noise distortion, speech distortion, and noise reduction is needed. The simple masking model used here was not successful at providing better performance for the hybrid or modified MMSE-LSA algorithms.

We were also concerned with how well single microphone speech enhancement techniques compare with multiple microphone techniques, in the context of a specific application to a cell phone. When the relevant acoustic systems are causally invertible, we are confident that an improvement can be made in the two-microphone case with the use of system identification techniques since an exact inverse of the MIMO system can be estimated. Unfortunately, there are a good deal of cases where nonminimum phase acoustic systems are present, which in turn results in tonal artifacts when attempting to causally reconstruct the speech. Furthermore, attempting to estimate the inverse of a nonminimum phase system composed with another system and using the causal result for reconstruction leads to poor noise suppression properties. The feasibility of any of these multiple microphone techniques in a real-time low-latency enhancement system is questionable, since long delays are required.

10.2 Future Work

Although we have shown improvements in enhanced speech quality for the single microphone speech enhancement case, these improvements are not dramatic. The difficulty resides in the noise adaptation scheme, which all the algorithms tested (except the IS-127 pre-processor) utilized. Any noise adaptation scheme operating in a STSA framework must make assumptions about the rate of change of the speech and noise spectra, and then make an estimate of the noise spectrum given the current frame of noisy speech and previous estimates of the speech and noise spectra. Whenever the spectrum of the noise changes more quickly than the noise adaptation scheme is capable of tracking, estimation errors result. This noise tracking bound places serious restrictions on the performance of the overall enhancement system.

A multi-channel speech enhancement solution avoids this problem of tracking highly non-stationary noise by making more accurate recordings of the noise available from different microphones. However, now the problem of understanding how the noise and speech at each microphone are related to each other is introduced, which depends on the acoustic environment of the system. Invertibility issues pose a serious treat to reconstruction of the original speech and noise signals. Note that invertibility is not the problem by itself. It is shown in [44] that the original signals can be exactly reconstructed by possibly adding more microphones to the system if all the relevant acoustic systems are known, even if the individual acoustic systems are nonminimum phase. We had difficulty with invertibility problems here because we were forced to factor the MIMO system in order to avoid systems that we are unable to estimate given the application requirements that all the enhancement equipment remain on the cell phone.

Our experience with the hybrid algorithm indicates that it makes a good aggressive noise suppressor. It performs very well, for instance, on low SNR helicopter noise, and might find an application better suited to its use elsewhere.

Our investigation has paid great attention to the quality of the enhanced speech, but has placed little emphasis on the delay and computational complexity of the various algorithms discussed. Both Malah's modified MMSE-LSA and the hybrid algorithm have a delay of 16 ms as implemented here, which compares quite unfavorably with the 3 ms of delay for Motorola's IS-127 pre-processor. Attempting to reduce the delay by simply decreasing the frame size adversely impacts the noise adaptation module so that initial fricatives are distorted.

The modified MMSE-LSA and hybrid algorithms as implemented here are necessarily more computationally complex than the IS-127 pre-processor, since they require the computation of larger DFTs, use a greater window overlap, and perform more elaborate gain and parameter update calculations. This difference could not be reliably measured since the modified MMSE-LSA and hybrid algorithms were written in Matlab, but the IS-127 pre-processor was optimized in C. We expect that work aimed towards reducing the delay and complexity of these algorithms will be fruitful in providing an effective and practical speech enhancement scheme.

References

[1] J. S. Lim, "Speech Enhancement," in *Prentice Hall Signal Processing Series*, A. V. Oppenheim, Ed., 1 ed. New Jersey: Prentice Hall, 1983, pp. 363.

[2] S. F. Boll, "Suppression of Acoustic Noise in Speech Using Spectral Subtraction," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-27, pp. 113-120, 1979.

[3] J. S. Lim and A. V. Oppenheim, "Enhancement and Bandwidth Compression of Noisy Speech," *Proceedings of the IEEE*, vol. 67, pp. 1586-1604, 1979.

[4] P. M. Crozier, B. M. G. Cheetham, C. Holt, and E. Munday, "Speech Enhancement Employing Spectral Subtraction and Linear Predictive Analysis," *Electronics Letters*, vol. 29, pp. 1094-1095, 1993.

[5] M. Lorber and R. Hoeldrich, "A Combined Approach for Broadband Noise Reduction," presented at Mohonk, Mohonk Mountain House, New Paltz, New York, 1997.

[6] Y. Ephraim and D. Malah, "Speech Enhancement Using a Minimum Mean-Square Error Short-Time Spectral Amplitude Estimator," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-32, pp. 1109-1121, 1984.

[7] Y. Ephraim and D. Malah, "Speech Enhancement Using a Minimum Mean-Square Error Log-Spectral Amplitude Estimator," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-33, pp. 443-445, 1985.

[8] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, 3 ed. New York: McGraw-Hill, Inc., 1991.

[9] R. M. Gray, A. Buzo, A. H. Gray, Jr., and Y. Matsuyama, "Distortion Measures for Speech Processing," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-28, pp. 367-376, 1980.

[10] L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals*, 3 ed. New Jersey: Prentice Hall, 1978.

[11] W. B. Kleijn and K. K. Paliwal, "Speech Coding and Synthesis," 1 ed. New York: Elsevier, 1995, pp. 755.

[12] J. S. Lim and A. V. Oppenheim, "All-Pole Modeling of Degraded Speech," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-26, pp. 197-210, 1978.

[13] J. D. Gibson, B. Koo, and S. D. Gray, "Filtering of Colored Noise for Speech Enhancement and Coding," *IEEE Transactions on Signal Processing*, vol. 39, pp. 1732-1742, 1991.

[14] R. J. McAulay and M. L. Malpass, "Speech Enhancement Using a Soft-Decision Noise Suppression Filter," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-28, pp. 137-145, 1980.

[15] O. Cappé, "Elimination of the Musical Noise Phenomenon with the Ephraim and Malah Noise Suppressor," *IEEE Transactions on Speech and Audio Processing*, vol. 2, pp. 345-349, 1994.

[16] T. F. Quatieri and R. A. Baxter, "Noise Reduction Based on Spectral Change," presented at Mohonk, Mohonk Mountain House, New Paltz, New York, 1997.

[17] J. A. Rice, *Mathematical Statistics and Data Analysis*, 2 ed. Belmont: Duxbury Press, 1995.

[18] Y. Ephraim and H. L. Van Trees, "A Signal Subspace Approach for Speech Enhancement," *IEEE Transactions on Speech and Audio Processing*, vol. 3, pp. 251-266, 1995.

[19] H. Fletcher, *Speech and Hearing in Communication*. New York: Acoustical Society of America, 1995.

[20] J. D. Johnston, "Transform Coding of Audio Signals Using Perceptual Noise Criteria," *IEEE Journal on Selected Areas in Communications*, vol. 6, pp. 314-323, 1988.

[21] J.-H. Chen and A. Gersho, "Adaptive Postfiltering for Quality Enhancement of Coded Speech," *IEEE Transactions on Speech and Audio Processing*, vol. 3, pp. 59-70, 1995.

[22] N. Jayant, "Signal Compression: Coding of Speech, Audio, Text, Image and Video," in *Selected Topics in Electronics and Systems*, vol. 9, P. K. Tien, Ed., 1 ed. New Jersey: World Scientific, 1997, pp. 231.

[23] E. Terhardt, "Calculating Virtual Pitch," *Hearing Research*, pp. 155-182, 1979.

[24] E. Terhardt, G. Stoll, and M. Seewann, "Algorithm for Extraction of Pitch and Pitch Salience from Complex Tonal Signals," *Journal of the Acoustical Society of America*, vol. 71, pp. 679-688, 1982.

[25] W. M. Hartmann, *Signals, Sound, and Sensation*, 1 ed. Woodbury: AIP Press, 1997.

[26] A. S. Willsky, G. W. Wornell, and J. H. Shapiro, *Stochastic Processes, Detection, and Estimation: Supplementary Course Notes*. Cambridge: MIT, 1996.

[27] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*, 8 ed. New Jersey: Prentice Hall, 1989.

[28] M. B. Priestley, *Spectral Analysis and Time Series*. New York: Academic Press, 1996.

[29] T. V. Ramabadran, J. P. Ashley, and M. J. McLaughlin, "Background Noise Suppression for Speech Enhancement and Coding," presented at IEEE Workshop on Speech Coding for Telecommunications, Pocono Manor Inn, Pocono Manor, Pennsylvania, 1997.

[30] E. J. Diethorn, "A Low-Complexity, Background-Noise Reduction Preprocessor for Speech Encoders," presented at IEEE Workshop on Speech Coding for Telecommunications, Pocono Manor Inn, Pocono Manor, Pennsylvania, 1997.

[31] T. Honkanen, J. Vainio, K. Järvinen, P. Haavisto, R. Salami, C. Laflamme, and J.-P. Adoul, "Enhanced Full Rate Speech Codec for IS-136 Digital Cellular System," Proc. *ICASSP '97*, Munich, pp. 731-734, 1997.

[32] M. R. Sambur and N. S. Jayant, "LPC Analysis/Synthesis from Speech Inputs Containing Quantizing Noise or Additive White Noise," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-24, pp. 488-494, 1976.

[33] P. Mermelstein and Y. Qian, "Nonlinear Filtering of the LPC Residual for Noise Suppression and Speech Quality Enhancement," presented at IEEE Workshop on Speech Coding for Telecommunications, Pocono Manor Inn, Pocono Manor, Pennsylvania, 1997.

[34] Y. Kaneda and J. Ohga, "Adaptive Microphone-Array System for Noise Reduction," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-34, pp. 1391-1400, 1986.

[35] E. Weinstein, M. Feder, and A. V. Oppenheim, "Multi-Channel Signal Separation by Decorrelation," *IEEE Transactions on Speech and Audio Processing*, vol. 1, pp. 405-413, 1993.

[36] B. Widrow, J. R. Glover, Jr., J. M. McCool, J. Kaunitz, C. S. Williams, R. H. Hearn, J. R. Zeidler, E. Doung, Jr., and R. C. Goodlin, "Adaptive Noise Cancelling: Principles and Applications," *Proceedings of the IEEE*, vol. 63, pp. 1692-1716, 1975.

[37] S. F. Boll and D. C. Pulsipher, "Suppression of Acoustic Noise in Speech Using Two Microphone Adaptive Noise Cancellation," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-28, pp. 752-753, 1980.

[38] E. Weinstein, A. V. Oppenheim, M. Feder, and J. R. Buck, "Iterative and Sequential Algorithms for Multisensor Signal Enhancement," *IEEE Transactions on Signal Processing*, vol. 42, pp. 846-859, 1994.

[39] M. Feder, A. V. Oppenheim, and E. Weinstein, "Maximum Likelihood Noise Cancellation Using the EM Algorithm," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, pp. 204-216, 1989.

[40] S. T. Neely and J. B. Allen, "Invertibility of a Room Impulse Response," *Journal of the Acoustical Society of America*, vol. 66, pp. 165-169, 1979.

[41] L. Ljung, *System Identification: Theory for the User*, 6 ed. New Jersey: Prentice Hall, 1987.

[42] A. V. Oppenheim and A. S. Willsky, *Signals and Systems*, 17 ed. New Jersey: Prentice Hall, 1983.

[43] R. O. Duda and W. L. Martens, "Range-Dependence of the HRTF for a Spherical Head," presented at Mohonk, Mohonk Mountain House, New Paltz, New York, 1997.

[44] M. Miyoshi and Y. Kaneda, "Inverse Filtering of Room Acoustics," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 36, pp. 145-152, 1988.