Linear and Nonlinear Loudspeaker Characterization

A Major Qualifying Project Report submitted to the faculty of

WORCESTER POLYTECHNIC INSTITUTE

in partial fulfillment of the requirements for the
Degree of Bachelor of Science
by

Samuel Brown

Date: _____

Approved: _____

Professor Donald R. Brown, Advisor

Abstract

This project developed both linear and nonlinear characterization techniques for use with dynamic loudspeakers. These techniques were designed to provide insight into the effectiveness of specifications typically used to describe loudspeaker transfer characteristics. Provisions were made for non-idealities present in the measurement environment. Final results included one-dimensional plots of linear speaker response, and two-dimensional plots of nonlinear response. Suggestions were made for future experimentation and development.

Table of Contents

Table of Figures

1. Introduction

The purpose of this project was to characterize several dynamic loudspeakers, via white noise excitation combined with several signal processing techniques. Both linear and nonlinear characterizations were used, and the results of the different methods compared. These tasks were undertaken with the intent of gaining insight into loudspeaker transfer characteristics that are not fully described with common technical specifications.

While specifications such as "frequency response" and %THD (Total Harmonic Distortion) are useful, they omit a significant amount of information. Unfortunately, these two descriptors are easily the most prevalent means of documenting loudspeaker transfer characteristics.

Frequency response is a purely linear measurement, so it does not take nonlinearities into account. There are several known nonlinearities present in dynamic loudspeakers, however. The extent to which these influence frequency response measurements, if at all, is a key concern in this report.

The lack of nonlinear information present in a frequency response measurement may be somewhat compensated for if it is paired with a %THD measurement. However, %THD is very unspecific. It simply delineates what percentage of total output energy is devoted to harmonic distortion. It does not illuminate which parts of the spectrum this energy occupies, and it does not account for inter-modulation distortion.

This report investigates alternatives to conventional measurement techniques and specifications. These alternatives better represent the actual transfer characteristics of dynamic loudspeakers, by taking into account both harmonic and inter-modulation distortion. The main contribution of this work is to create a loudspeaker model that contains both fist- and second-order transfer functions, which can be represented and meaningfully interpreted in a visual manner.

2. Background

Of all commonly accepted practices for specifying loudspeaker characteristics, frequency response is by far the most pervasive. However, frequency response alone is hardly sufficient to describe the on-axis transfer characteristics of real-world speakers. The measurement itself is based upon the assumption that loudspeakers perform a linear transformation, which is an ideal that is never perfectly achieved in practice.

Since frequency response is a purely linear measurement, it fails to take into account the known, nonlinear transfer characteristics of loudspeakers. Furthermore, nonlinear distortion in speakers is typically described with a simple percentage, i.e. %THD (Total Harmonic Distortion). Since distortion is the most sonically displeasing characteristic of loudspeakers [3], more in-depth measurements would prove invaluable by providing more detailed information about such nonlinearities.

In light of the above information, the background section of this report will treat principles of loudspeaker operation, methods of linear and nonlinear characterization, and various techniques for the analysis of the involved measurements. Since this report is strictly concerned with physical phenomena, all linear and nonlinear systems under discussion are assumed to be causal.

2.1 Basic principles of loudspeaker operation

The basic construction of a dynamic loudspeaker consists of any number of "drivers" mounted on a flat surface called a "baffle" and placed within an enclosure. Each driver consists of a flexible cone attached to a coil of wire that is mounted so that it can move freely, within limits, inside of a fixed magnet. This coil is referred to as the "voice coil." Electrical currents passing through the coil create a varying magnetic field, which reacts with the fixed field and produces mechanical fluctuations of the coil. The cone then moves in turn, and sets a column of air in motion both in front of and behind the cone. In this way, an electrical signal is converted into a sound pressure wave [3]. A cutaway view of a typical dynamic loudspeaker driver is shown in Figure 1.

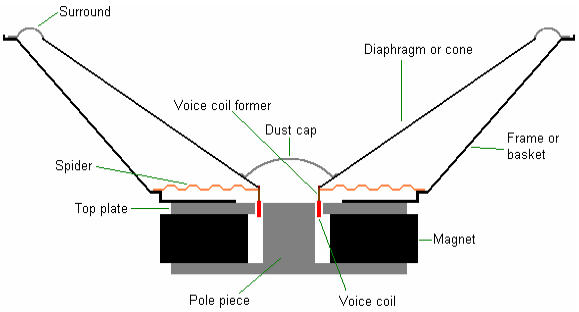

Figure 1. Construction of a dynamic loudspeaker driver [18].

It is difficult to reproduce the entire range of audible frequencies using a single driver, so many loudspeakers have separate low- and high-frequency drivers mounted in a single enclosure [12]. These are commonly referred to as "two-way" systems; systems with even more drivers are also available. The high-frequency drivers are often of a different construction (e.g., electrostatic rather than dynamic), and the details of such alternative driver models can be found in [3] and [12]. Since it is mid- to low-frequency drivers that produce the most prominent and psychoacoustically objectionable distortions and nonlinearities [3], they are of primary concern in this report.

Ideal loudspeakers produce acoustic waves that are a linear transformation of the electrical input (excitation) signal [8]. This fact is the underlying assumption in traditional, "frequency response" analysis of electro-acoustic systems, wherein multiple FFT measurements are used to generate an average system response in the frequency domain. While this assumption relies on obvious oversimplification of the electro-mechanical properties of typical dynamic loudspeakers, such analysis techniques still provide useful insight into speaker performance.

When considering the real-world frequency response of a loudspeaker, several contributing factors are linear in nature. Reflections off of the enclosure can create either constructive or destructive interference with the direct sound emanating from the speaker cone, leading to peaks and dips in the response. Refraction around the edges of the baffle can create decreased sensitivity at higher frequencies. Resonances in the speaker assembly and/or enclosure may create increased sensitivity at discrete lower frequencies. Uncompensated, frequency-dependent impedances in the associated circuitry also lead to

uneven sensitivity [12]. All of these factors consequently manifest themselves in the linear transfer characteristic of a loudspeaker; namely, its frequency response.

The nonlinearities present in a loudspeaker may be generated by the circuits through which the audio signal passes, or by the mechanical components responsible for electro-acoustical transduction. This distortion takes on many forms. Most forms are unpleasant to hear if present in large amounts, though small percentages are usually tolerated, or even masked, by the human ear [3].

Electrical distortion in loudspeakers is often produced at high output levels, caused by the limited dynamic range of any associated circuitry. The self-inductance of the voice coil, however, also produces electrical distortion. Mechanical distortion, which is exacerbated at high output levels, is produced by: (i) hysteresis in the rim of the speaker cone, (ii) limited voice coil excursion, which may cause clipping and "bottoming" effects, and (iii) subharmonics generated by the loudspeaker cone [11].

The two most prominent and well-documented forms of distortion in loudspeakers are: (i) harmonic distortion, and (ii) inter-modulation distortion. These two nonlinearities are treated below.

Harmonic distortion – Harmonic distortion is characterized by the presence of harmonics in the output not present in the original (excitation) signal. Usually, harmonic distortion is created by insufficient dynamic range, but any nonlinearity in a transfer function – electrical or mechanical – can also create harmonics [3]. When program material is used as the excitation signal, this type of distortion is often unnoticeable because the harmonics are either masked, or simply add to those already present in the signal. Depending on how the harmonics add to the original signal, sometimes a subtle amount of harmonic distortion is pleasing to the human ear.

Inter-modulation distortion – Inter-modulation distortion arises whenever two signals at different frequencies pass simultaneously through a system with a nonlinear transfer characteristic. The lower frequency modulates the higher frequency, producing two new frequencies that are the sum and difference of the original input frequencies. Harmonics of the input frequencies are also available for modulation, and the original input frequencies may even modulate their harmonics, thus creating a chain reaction up through the audible spectrum [3]. In this way, linear combinations of the fundamental frequencies and all harmonics present in the input signals may appear at the loudspeaker output.

When a complex signal such as program material is used to excite a speaker, the number of inter-modulation terms quickly becomes astronomical. Most of these terms are very small, however, and can be ignored. This is fortunate because, to the human ear, inter-modulation distortion is the most offensive loudspeaker nonlinearity [3].

When considered simultaneously, the linear and nonlinear transfer characteristics of loudspeakers form a complete system model. An accurate description of speaker performance must therefore incorporate both.

2.2 Measurement Techniques

There has been significant advancement in the field of electro-acoustics since its inception, but the basic testing setup for loudspeakers has remained largely unchanged. In a typical setup, the speaker is excited with a known audio signal, which then passes through an acoustic channel (which may or may not add noise to the signal), and is captured by a measurement microphone. The output of the microphone is then processed immediately and/or recorded for future analysis [6]. This typical setup is illustrated in Figure 2.

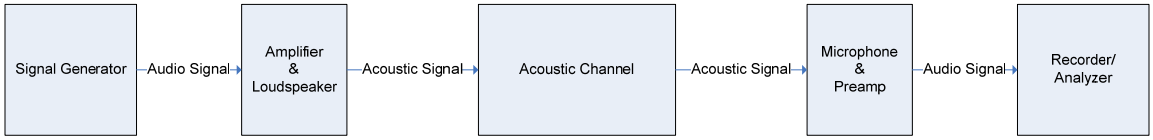

Figure 2. Loudspeaker testing setup.

When measuring loudspeaker characteristics, there are two important factors that must be carefully considered in order to provide meaningful results: (i) the physical conditions under which the measurements are performed, and (ii) the instrumentation that is used. These factors are treated below, in that order.

Physical considerations include the acoustical properties of the measurement environment, and the placement of any testing equipment. In general, the environment should influence the results as little as possible, and the loudspeaker and microphone placement should be tailored to minimize any non-idealities therein. All other equipment should be placed as far from the testing setup as is practical.

Acoustic channel – The first and foremost condition of measurement is the acoustic environment in which testing takes place. The standard environment for loudspeaker measurements is an anechoic chamber, which approximates free-field conditions, even though locations of normal loudspeaker operation do not often correspond to the free-field. This is because measurements taken in the free-field have the advantage of being well defined, repeatable, and allow the comparison of results with relative ease [6]. When an anechoic chamber is not available, tests are sometimes performed outdoors. For frequency response calculations, pseudo-anechoic measurement techniques are also available (though these will not be addressed in this report).

Loudspeaker mounting – The second condition of measurement that must be considered is the manner in which the loudspeaker is mounted within the test environment. When an anechoic chamber is used, the speaker is suspended at a position that takes advantage of the properties of that particular space. This position naturally varies between chambers. When measurements are performed

outside, a free-field may be simulated by elevating the speaker a sufficient distance above the ground, depending upon how much attenuation of ground reflections is desired.

Alternately, a "half-space" measurement may be taken by placing the speaker on the ground with its cone facing the sky. Ground reflections will still interfere, but will be significantly reduced because most loudspeakers are directional, and only radiate low frequencies backwards [19]. In this way, irregularities in midrange frequency response calculations caused by ground reflections may be nearly eliminated. The low frequencies that are reflected will be in-phase with the direct sound, due to their longer wavelengths, and this will create a virtual bass boost in any measurements. However, since speakers are often placed against a wall, this corresponds to typical listening conditions.

Microphone placement – The last condition of measurement is the positioning of the microphone relative to the loudspeaker. Unless directionality information is desired, all testing should be performed with the microphone directly on the axis of the speaker [6]. The microphone should also be placed level with the high frequency driver if the loudspeaker under test is a two-way system [19]. This is due to the increased directionality of higher audio frequencies. A measuring distance of 1 to 2 meters is generally accepted as appropriate, and minimizes phase cancellation between multiple drivers [6], [19].

In addition to the loudspeaker under test, measurement instrumentation includes a signal generator of some sort, a power amplifier for the speaker, a measurement microphone and preamp, and any analysis and/or recording equipment (refer to Figure 2). It is important to understand both the electrical and acoustical properties of this instrumentation, in order to provide repeatable results that can be compared with those from other parties.

Signal generator – The signal generator typically falls into one of three categories: (i) an oscillator, (ii) a generator of some other deterministic signal, or (iii) a known random signal generator. The most common signals used for loudspeaker transfer function measurements are sinusoidal or Gaussian in nature. Now that high-quality, low-cost soundcards are readily available, computers are often used to generate any of the above described signals.

Power amplifier – The operation of the power amplifier is straightforward, as it simply amplifies the output of the signal generator in order to deliver sufficient power to drive the loudspeaker. The amplifier should provide nominally flat response between 20 Hz and 20 kHz, and generate a minimum of harmonic distortion.

Measurement microphone – Proper microphone selection is critical to accurate measurements. There are many different types currently available for a plethora of applications, but the only proper microphones for electro-acoustic measurements are those designed specifically for that purpose. In addition to extremely flat response across all audible frequencies, these microphones have a very small profile. Their flat frequency response and small size allow for a minimal influence on any measurements.

Microphone preamp – The microphone preamp is as straightforward as the power amplifier for the loudspeaker. It simply amplifies the microphone output to a level suitable for recording. As with the power amplifier, it should have flat response and negligible harmonic distortion.

Signal recorder/analyzer – The last instrument in the measurement signal chain is the recorder/analyzer. This may be an oscilloscope, spectrum analyzer, analog or digital recorder, or any number of specialized pieces of hardware. As with signal generation, recording and analysis are now often performed on PCs due to the growing availability of quality, economical soundcards. This poses the

additional advantage of nearly unlimited post-processing options. Using PCs for both signal generation and recording/analysis also eases synchronization between reference signals and their measured counterparts, and reduces the amount of necessary equipment.

Regardless of the loudspeaker characteristic being measured, the test setup remains generally consistent. Of all the elements of the signal chain outlined in Figure 2, the properties of the signal generator and recorder/analyzer vary the most between different types of measurements. For simplicity's sake, the remainder of this report will assume that both of these elements are replaced by a soundcard/PC combination. This allows for a number of different testing techniques.

2.3 Linear Characterization of Loudspeakers

A linear, time-invariant system with time-domain excitation signal *x(t)* and time-domain output *y(t)* is shown in Figure 3. This system is representative of an ideal loudspeaker, where *x(t)* is an electrical signal and *y(t)* is a sound pressure wave.

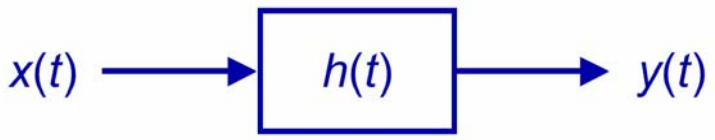

Figure 3. Model of a linear system [20].

Given the above linear system, it can be shown that

(1) $y(t) = x(t) \cdot h(t)$,

where *h(t)* is the impulse response of the system. Denoting *X*(*f*), *Y*(*f*), and *H*(*f*) as the frequency-domain representations of *x(t)*, *y(t)*, and *h(t)*, respectively, (1) may be rewritten as

(2) $H(f) = \frac{Y(f)}{X(f)}$

One may determine the frequency response of the linear system by calculating *H(f)* at *M* distinct frequencies $f_k, k = 1,2,..., N/2$ over bandwidth *f* in measurement interval τ. The

measurement period *p* is the total time spent attempting to characterize the system, and may encompass a large number of measurement intervals [23].

For loudspeakers, the most useful application of frequency response is simply determining the range of audio frequencies a speaker is capable of reproducing, and with what general level of consistency. Loudspeakers whose sensitivity is largely consistent across the range of audible frequencies are said to have a "flat" frequency response. This is commonly accepted as the ideal, though it is important to remember that frequency response does not fully describe the transfer characteristics of any electro-acoustic device.

When measuring the linear response of a loudspeaker, several techniques are available. These are all determined by the type of excitation signal used. Two common techniques that are particularly well-suited to solving (2) are treated below: (i) a stepped sine sweep, and (ii) White noise excitation. For a detailed explanation of other popular (and more involved) methods, refer to [2], [24].

Stepped sine sweep – When a stepped sine measurement is performed, a signal of the form $x(t) = A\sin(2\pi \cdot k \cdot t)$ is used to excite the loudspeaker under test. For each discrete frequency k=1,2,…,M, the response of the speaker is measured. In this way, the total frequency response over bandwidth f may be calculated. The measurement period p is the set of M measurement intervals τ, one for each frequency [23].

White noise excitation – White noise is a Gaussian random signal with constant power density, which contains all M frequencies in bandwidth f. When a loudspeaker is excited with such a signal, the response can be accurately measured by averaging the results obtained from performing FFT analysis over several measurement intervals τ [23].

In general, there are numerous tradeoffs between sine-based and noise-based electro-acoustic measurements. White noise excitation is particularly well-suited to the measurement of loudspeaker transfer characteristics because it more closely represents the program material that is played through speakers in real-world applications [4]. However, noise-based FFT techniques suffer from spectral leakage [14], and are more susceptible to acoustic noise from other sources in the measurement environment [24] than their sweep-based counterparts.

The heightened susceptibility of noise-based measurements to a noisy acoustic channel is of particular concern in this report. This is because the methodology herein utilizes white noise excitation exclusively, and ideal facilities were unavailable at the time of measurement.

The model of Figure 3 may be modified as shown in Figure 4, in order to account for the presence of noise.

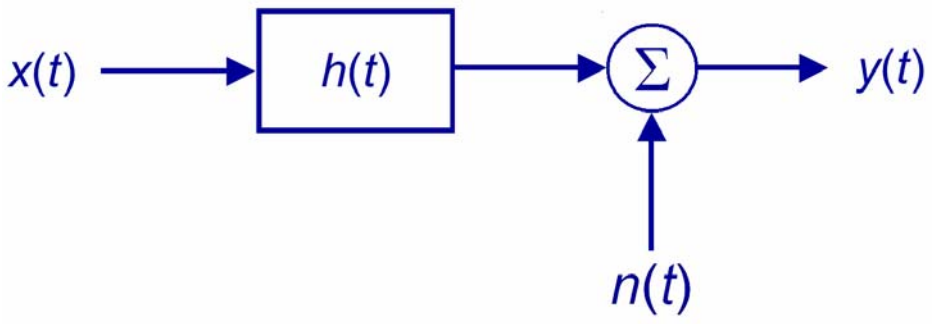

Figure 4. Model of a linear system with added noise [23].

The accuracy of transfer function measurements when the measurement is contaminated by noise may be improved with either the *H1* or *H2* method [23]. If the noise signal *n(t)* in Figure 4 is uncorrelated with *x(t)*, then the amplitude accuracy of any measurements may be improved by using the following equation to calculate *H(f)*:

$$(3) \qquad H1(f) = \frac{G_{xy}}{G_{xx}},$$

where G_{xx} and G_{xy} are the auto- and cross-spectra of *X(f)* and *Y(f)*, respectively [25]. This is because the noise is averaged out in G_{xy} [28].

Transfer function measurements in the presence of noise are further degraded when the spectrum of the excitation signal is sparse. All real-world excitation signals may be considered sparse, to varying degrees – even "Gaussian" signals produced by signal generators [23]. When considering the calculations outlined in Section 2.1, a "sparse" signal is one such that for a given measurement interval τ, not all M distinct frequencies over bandwidth f are present in the signal [23]. It follows that the division of (2) would fail for every f_k not present in $X(f)$. In practice, however, leakage from adjacent frequency bins means $X(f_k)$ is never exactly zero [14]. Nonetheless, measurements lose precision when $X(f_k)$ is very small, and are simply wrong when $N(f_k)$ is significantly larger than $X(f_k)$.

If all M discrete frequencies within bandwidth f are present in $x(t)$ for at least one measurement interval τ during measurement period p, then accuracy can be improved by averaging the FFT results from each interval. However, the average will still be contaminated by transfer functions computed when $X(f_k)$ is small and $N(f_k)$ is large. This may be counteracted by using the *H1* method given by (3), though there are several more techniques for improving amplitude accuracy under sparse excitation conditions. Two of these techniques are treated below: (i) signal thresholding, and (ii) coherence blanking.

Signal Thresholding – With signal thresholding, an amplitude threshold is set for $X(f_k)$; $Y(f_k)$ is then gathered for several measurement intervals and whenever $|X(f_k)|$ exceeds the threshold, the division of (2) is performed. If all M frequencies are present in $x(t)$ for at least one measurement interval, then the measured frequency response will be identical to one obtained using perfect Gaussian noise excitation and traditional FFT analysis [23].

Signal thresholding serves to significantly improve the signal-to-noise ratio of any results, because measurements are ignored when $x(t)$ is below a set threshold. Also, if two or more samples of the transfer function $H(f)$ are obtained over the

measurement period, then these may be averaged together. This reduces noise contamination even further.

Coherence Blanking – Coherence blanking similarly excludes measured data from transfer function calculations if the coherence between samples degrades over several measurement intervals, and fails to meet certain criteria. The coherence is given by:

$$(4)\ \text{coherence} = \gamma^2 = \frac{|G_{xy}|^2}{G_{xx} \cdot G_{yy}},$$

where *Gxy*, *Gxx*, and *Gyy* are averaged over several measurement intervals [23]. Note that the calculation of (4) is equivalent to the output power due to input, divided by the total output power.

Given (4), the trace representing the transfer function may be blanked at each frequency bin where the coherence drops below a preset threshold. In this way, contamination caused by uncorrelated noise is reduced. Note that this thresholding operates on γ^2, not $|X(f_k)|$ as with signal thresholding. The coherence threshold is dependent on the number of samples being considered, and suggested values can be found in [23].

By using the *H1* method combined with signal thresholding and coherence blanking, the effect of noise on linear transfer function measurements may be significantly reduced.

2.4 Nonlinear Characterization of Loudspeakers

Forgoing a rigorous derivation of a lumped-parameter loudspeaker model (which can be found in [15] and [17]), it is sufficient for this report to understand that loudspeaker transfer characteristics contain nonlinearities that result in added harmonics, and linear combinations thereof. This makes loudspeaker transfer functions particularly well-suited for representation by a Volterra series expansion, which may be used to represent time-invariant nonlinear systems.

The Volterra series is a form of the Taylor series that is extended to account for higher-order polynomial terms and system memory [27]. Using a Volterra series, the response of a discrete-time system may be written in the form:

$$(5) \qquad y[t] = \sum_{i_1=0}^{N-1} h_1[i]x[t-i_1] + \sum_{i_1=0}^{N-1}\sum_{i_2=i_1}^{N-1} h_2[i_1,i_2]x[t-i_1]x[t-i_2] + \sum_{i_1=0}^{N-1}\cdots\sum_{i_m=i_{m-1}}^{N-1} h_m[i_1,\ldots,i_m]x[t-i_1]\ldots x[t-i_m],$$

where $x[t\text{-}i_m]$ is a delayed version of the system input $x[t]$ at time t, and $h_m[i_1,\ldots,i_m]$ is the m^{th}-order kernel. Each kernel is a generalized impulse response, and the first term of (5) represents the convolution for a linear system [15].

Since the output $y[t]$ of (5) is linear with respect to the kernels, linear filter theory may be conveniently applied to system identification and analysis. Denoting $X[f]$ and $Y[f]$ as the discrete frequency-domain representations of *x(t)* and *y(t)*, respectively, the first two terms of (5) may be rewritten as

$$(6) \qquad Y[f_k] = H_1[f_k]X[f_k] + \sum_{\substack{i=-(M-1) \\ i+j=k}}^{M} \sum_{j=-(M-1)}^{M} H_2[f_i, f_j]X[f_i]X[f_j],$$

where $H_1[f_k]$ and $H_2[f_i,f_j]$ are the linear and quadratic transfer functions given at a discrete set of frequencies $f_k = k/N, k = -N/2+1,\ldots,-2,-1,0,1,2,\ldots,N/2$, and $f_M=f_{N/2}$ signifies the Nyquist frequency associated with the sampling of the time domain signals [16].

One may rewrite (6) as

$$(7) \qquad \begin{aligned} Y[k] = {} & H_1[k]X[k] \\ & + \underbrace{H_2[k-M,M]X[k-M]X[M] + \ldots + H_2[-1,k+1]X[-1]X[k+1] + H_2[0,k]X[0]X[k]}_{(a)} \\ & + \underbrace{H_2[1,k-1]X[1]X[k-1] + \ldots + H_2[k-1,1]X[k-1]X[1]}_{(b)} \\ & + \underbrace{H_2[k,0]X[k]X[0] + H_2[k+1,-1]X[k+1][-1] + \ldots + H_2[M,k-M]X[M]X[k-M]}_{(c)} \end{aligned}$$

In equation 7, the terms in *(a)* and *(c)* represent the difference interactions of the frequency components in the input signal, and the terms in *(b)* are due to the sum interactions [16]. Thus this model clearly covers both second-order harmonic and inter-

modulation distortion, the two most prominent forms of nonlinear distortion in loudspeakers [5].

Inspection of (7) yields that the output portion of (a) is identical to (c), due to the symmetry of the quadratic transfer function [16]. This allows one to reduce the number of quadratic kernels by roughly ½. Consequently, one may significantly reduce the computational complexity of simultaneously solving (6) for $H_1[f_k]$ and $H_2[f_i,f_j]$.

Using vector notation, one may also rewrite (7) as

(8) $$Y[k] = H_v[k]^T X_v[k] = X_v[k]^T H_v[k],$$

where the Volterra kernel vector H_v contains the linear and quadratic transfer functions, and the vector X_v contains the linear and quadratic inputs [16]. These two vectors are shown below:

(9) $$H_v[k] = \left[H_1[k], H_{2v}[k]\right], \text{ and}$$

(10) $$X_v[k] = \left\lfloor X[k], X_{2_v}[k]\right\rfloor,$$

where the vectors containing the respective quadratic transfer functions and inputs, $H_{2_v}[k]$ and $X_{2_v}[k]$, are indexed as follows:

(11) $$H_{2_v}[k] = \left[2H_2\left[\frac{k+1}{2},\frac{k-1}{2}\right],2H_2\left[\frac{k+3}{2},\frac{k-3}{2}\right],...,2H_2\left[\frac{M}{2},k-\frac{M}{2}\right]\right], \text{ for odd } k, \text{ and}$$

$$H_{2_v}[k] = \left[H_2\left[\frac{k}{2},\frac{k}{2}\right],2H_2\left[\frac{k}{2}+1,\frac{k}{2}-1\right],...,2H_2\left[\frac{M}{2},k-\frac{M}{2}\right]\right], \text{ for even } k;$$

(12) $$X_{2_v}[k] = \left[X\left[\frac{k+1}{2}\right]X\left[\frac{k-1}{2}\right], X\left[\frac{k+3}{2}\right]X\left[\frac{k-3}{2}\right],..., X\left[\frac{M}{2}\right]X\left[k-\frac{M}{2}\right]\right], \text{ for odd } k, \text{ and}$$

$$X_{2_v}[k] = \left[X\left[\frac{k}{2}\right]X\left[\frac{k}{2}\right], X\left[\frac{k}{2}+1\right]X\left[\frac{k}{2}-1\right],..., X\left[\frac{M}{2}\right]X\left[k-\frac{M}{2}\right]\right], \text{ for even k.}$$

The optimum solution for $H_v[k]$ in (8), under the minimum mean-squared-error (MMSE) criterion, is then given by

$$(13) \qquad H_v[k] = \mathrm{E}\left\{X_v^*[k]X_v^T[k]\right\}^{-1}\mathrm{E}\left\{X_v^*[k]Y[k]\right\},$$

as described in detail in [5], [16]. When a sufficient number of measurement intervals are obtained, so that the expectation operators may be replaced with ensemble averages, then (13) is reduced to a simple least-squares operation [29].

When measuring the nonlinear response of a loudspeaker, there are two techniques available to solve (6) for $H_1[f_k]$ and $H_2[f_i,f_j]$: (i) stepped sinusoidal excitation, and (ii) White noise excitation. These two techniques are briefly treated below.

Stepped sinusoidal excitation – When a stepped sine/multi-sine measurement is performed, the harmonic and inter-modulation distortions over bandwidth *f* are determined by repeating the same experiment at various frequencies and by changing the corresponding frequencies of oscillators and filters [5]. This method is very tedious, does not apply to the measurement techniques outlined in this section, and is impractical for anything but very small values of *M*.

White noise excitation – As stated previously, white noise is a Gaussian random signal with constant power density, which contains all *M* frequencies in bandwidth *f*. When a loudspeaker is excited with such a signal, the distortions can be estimated simultaneously across *f* using (13), as long as the data set is sufficiently large to replace the expectation in (13) with an ensemble average over several measurement intervals τ [23].

White noise excitation is clearly the superior option when measuring the nonlinear transfer characteristics of loudspeakers. However, as with linear characterization, this technique's susceptibility to non-idealities in the acoustic channel must be taken into account. It is important to note that the noise-reduction techniques outlined in Section 2.3 cannot be applied to nonlinear analysis, due to their reliance on first-order coherence spectra. Nonlinear results are therefore influenced more heavily by the non-idealities of the measurement environment.

3. Methodology

The methodology section of this report discusses the loudspeakers that were tested, and how measurements were obtained. This process consists of the physical measurement of loudspeaker output under excitation by a reference signal, followed by appropriate signal processing – of both reference and measured signals – to obtain the desired results.

3.1 Loudspeakers tested

All of the loudspeakers chosen for testing were powered. This simplified the testing procedure by reducing the amount of hardware under consideration – the power amplifier and loudspeaker could be treated as a single unit. The speakers under test are shown below.

Figure 5. M-Audio Studiophile SP-5B.

Figure 6. Alesis M1 Active MkII.

Figure 7. Yamaha MSP5.

For detailed technical specifications of the M-Audio, Alesis, and Yamaha loudspeakers, refer to [21], [1], and [31], respectively. All of these speakers demonstrate a nominal frequency response of ±5 dB from 100 Hz to 20 kHz (or better).

3.2 Measurement Equipment

The microphone used in all tests was a B&K Type 4006 omnidirectional condenser, a model designed specifically for performing critical electro-acoustic measurements. This microphone is shown in Figure 8.

Figure 8. B&K Type 4006.

The B&K Type 4006 demonstrates a factory-measured frequency response of ±2dB from 20 Hz to 20 kHz. For detailed technical specifications, refer to [9].

A Tascam US-122 USB soundcard was used to interface the speakers and microphone with the laptop. This soundcard is shown in Figure 9. The US-122 allows for full-duplex operation over stereo inputs and outputs, and has the additional benefit of built-in microphone preamps. Detailed technical specifications for this soundcard can be found in [30].

Figure 9. Tascam US-122 USB soundcard.

The inputs and outputs of the US-122 were assigned as shown in Table 1. By feeding the mono excitation signal back to one channel of the stereo input, the transfer characteristic of the pre-amp could be compensated for by using this fed-back signal as a reference (rather than the excitation signal itself).

Table 1. Soundcard channel assignments.

Soundcard Channel		**Assignment**
Outputs	US-122 Output – Left	Loudspeaker input
	US-122 Output – Right	US-122 Input – Left
Inputs	US-122 Input – Left	US-122 Output – Right
	US-122 Input – Right	Microphone output

In order to streamline the measurement process, a laptop was used for signal generation, recording, and analysis. All of these processes were carried out in software. MATLAB was used to generate a Gaussian noise excitation signal, recording was performed using Cakewalk SONAR 2.0, and MATLAB was again used to analyze the recorded signals.

3.3 Measurement procedure

Neither an anechoic chamber nor a quiet outdoor location was available for testing, so all measurements were taken in an acoustically treated listening room. This may have made the measurements more difficult to repeat; however, it afforded an environment that closely replicated real-world loudspeaker operating conditions. The treatments in the room also minimized early reflections and room modes, thus significantly reducing the effect of the room's impulse response on the measurements.

In order to compensate further for room reflections, each loudspeaker under test was placed on the floor in the center of the room, with its cone facing the ceiling. It was therefore located as far as possible from all reflecting surfaces except for the floor. In this way, the "half-space" measurement setup described in Section 2.2 was approximated using an indoor environment.

For each test, the microphone was placed on-axis, approximately 1 meter above the loudspeaker and aligned directly with the high-frequency driver. It was held in place at the end of a boom, so that the microphone stand could be placed away from the loudspeaker. This reduced the effect that any reflections off of the stand had on the measurements.

The complete test and measurement signal chain is shown in Figure 10. This particular testing setup was modeled after the generalized one – shown in Figure 2 – that was outlined in Section 2.2.

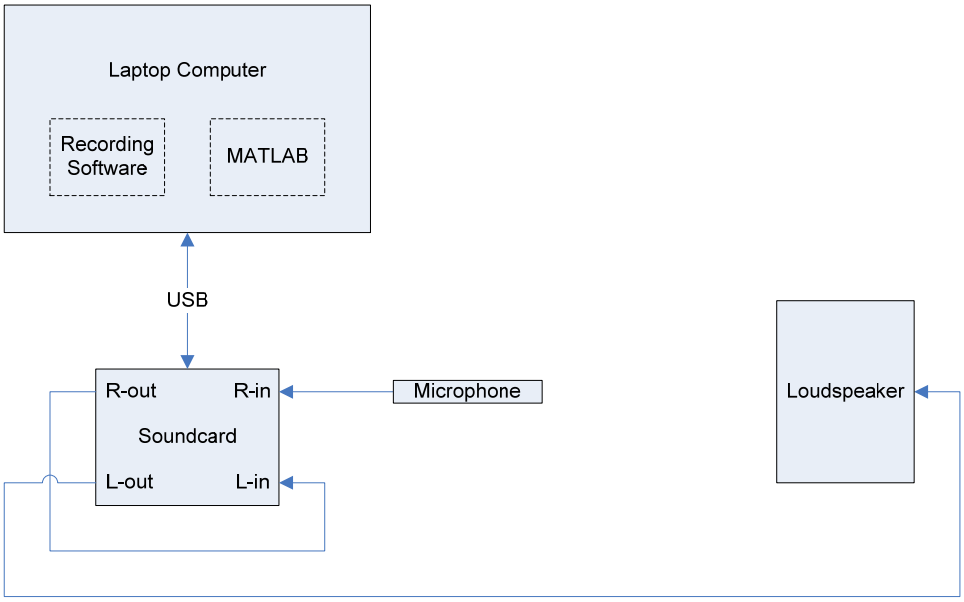

Figure 10. Test and measurement signal chain.

White noise excitation was selected for characterization because the corresponding measurement techniques are straightforward and, as stated in Section 2.2, it closely represents the program material that is played through speakers in real-world applications. Each loudspeaker under test was mounted on the floor then excited with 30 seconds of white nose, and both reference and measured signals were recorded. A length of 30 seconds was chosen because it was much longer than the impulse response of the room, without being so long as to make the testing procedure arduous and inconvenient.

3.4 Linear Characterization

After reference and measured signals were recorded for each speaker, these signals were saved as .wav files and loaded into MATLAB. They were then divided into M overlapping segments of length P, with overlap length $P/2$. Before performing any calculations, a symmetric Hann window was applied to each segment. After windowing, the following data were generated for each segment:

1. *xcorrx, ycorry, ycorrx*: auto- and cross-correlation functions of length $2P$-1
2. *Gxx, Gyy, Gyx*: auto- and cross-spectra of length N
3. *X, Y*: Fourier spectra of length N
4. γ^2: a coherence function, given by (3), of length N

Each data set was organized in a matrix whose columns corresponded to measurements, and whose rows corresponded to each segment, or measurement period.

The specific values for M, N, and P were of particular concern. Since 30 seconds of white noise were used for excitation, the number of segments M was solely dependent on the size of the measurement period P. Through experimentation, the value of P was chosen to obtain the maximum number of measurements without unduly sacrificing frequency resolution. The FFT length N could then be set at a value equal to P, or greater if zero padding was used. For specific values, refer to Section 4.

Once all of the above data had been generated, signal thresholding could be performed. A separate threshold was generated for each frequency bin in Gxx, by scanning the values in that bin across all measurement periods, then setting the threshold a set distance below the maximum value. Any bins that fell below that threshold were zeroed out.

Coherence blanking was performed in a similar manner. However, the value of the coherence function at each bin, rather than the value in that bin, determined whether it would be zeroed out or not.

Once signal thresholding and coherence blanking were completed, the frequency response of the loudspeaker could be calculated. First, the number of non-zero values for each bin was determined. This number was then used to average the response across all measurement periods, and the division of (3) was performed at each non-zero bin. In this way, a transfer function measurement with a greatly improved signal-to-noise ratio was determined.

As a final step, 1/3-octave smoothing was performed. This was appropriate because it is generally accepted as approximating the sensitivity of the human ear [7].

3.5 Nonlinear characterization

The first step in nonlinear characterization was to solve (8) for $H_v[k]$, by applying the minimum mean-squared-error method, given by (13), to each frequency bin. This was accomplished one bin at a time, in the following manner:

1. The frequency index, $m = 0...N/2$, was fixed at the current value
2. The matrix $\boldsymbol{M}_1$, used to store the average value of $X_v^*[m]X_v^T[m]$, was initialized to zeros
3. The matrix $\boldsymbol{M}_2$, used to store the average value of $X_v^*[m]Y[m]$, was initialized to zeros
4. For index $i = 1...M$ (the number of measurement periods)
    a. $X_v[m]$ was constructed from $X[k]$, according to (12)
    b. $X_v^*[m]X_v^T[m]/M$ was added to $\boldsymbol{M}_1$
    c. $X_v^*[m]Y[m]/M$ was added to $\boldsymbol{M}_2$
5. The value of $H_v[m]$ was set to $\boldsymbol{M}_1^{-1}\boldsymbol{M}_2$

In the case of nonlinear characterization, the measurement interval *P* was set to the same value as *N*, which was selected to provide the maximum frequency resolution without using too much computer memory to calculate $H_v[k]$. The choice to set *P*=*N* was made because zero-padding would create redundancy in *X*[*k*], thus creating poor Eigen values for $\boldsymbol{M}_1$.

Once the above had been completed for every frequency bin, the linear and quadratic transfer functions of (6) were constructed from $H_v[k]$, according to (11). As with linear characterization, 1/3-octave smoothing was performed on the results.

4. Results

This section of the report will treat the results of both the linear and nonlinear measurements described in Section 3. The effects of the signal processing techniques used to generate these results will be described in detail. Possible inaccuracies, largely caused by the noise present in the measurement environment, will also be described.

4.1 Linear results and analysis

The linear results section will outline the effects of each signal processing method before presenting the final, ensemble results. Basic measurements, without any processing, will be discussed first.

4.1.1 Basic results

Basic frequency response measurements of the Alesis speaker, calculated using both (2) and (3), are shown in Figure 11. A measurement period *P* of 1024 samples and a 1024-point FFT (*N*=1024) were used.

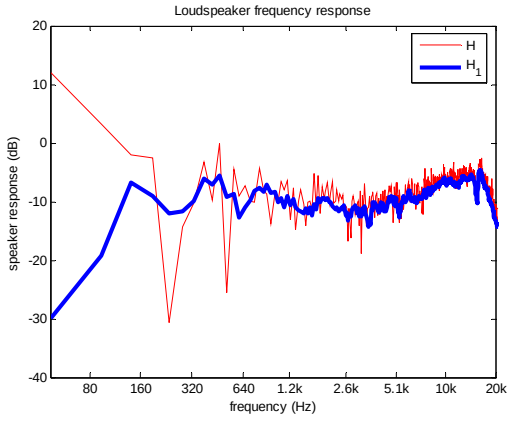

Figure 11. Basic loudspeaker frequency response, 1024-point sampling period.

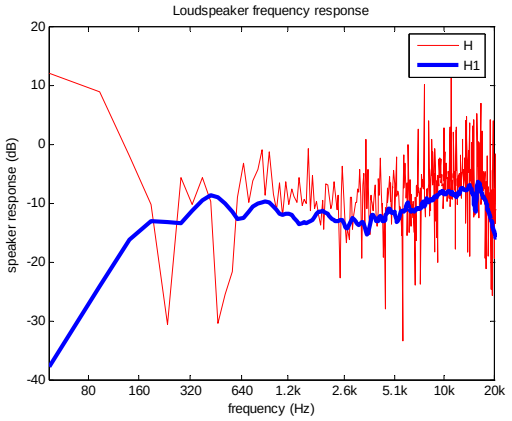

Figure 12. Basic loudspeaker frequency response, 512-point sampling period.

Inspection of Figure 11 shows that the *H1* method significantly reduces the amount of noise present in the results, especially in the lower frequencies. This noise reduction becomes even more obvious when the measurement period is reduced to 512 samples, and zero-padding is used to maintain a 1024-point FFT. This is shown in Figure 12.

The conventional FFT method, given by (2), generates much noisier results in Figure 12 than it does in Figure 11. The *H1* method is even smoother, however, because the smaller sampling period allows for a greater number of averages per bin.

4.1.2 Signal Thresholding

The same measurements of Figure 11, with a signal threshold set 24 dB below the maximum value in each bin, are shown in Figure 13.

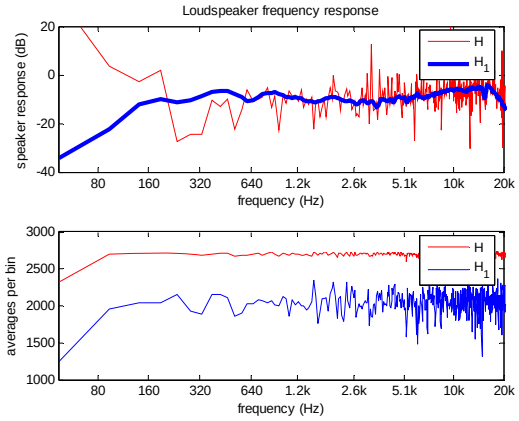

Figure 13. Loudspeaker frequency response with -24 dB signal threshold.

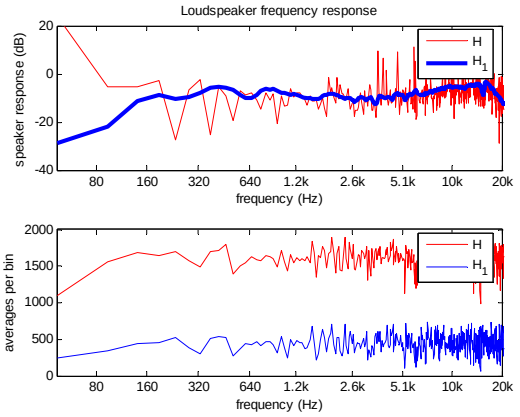

Figure 14. Loudspeaker frequency response with -12 dB signal threshold.

Inspection of Figure 13 shows that the *H1* method benefits from signal thresholding more than the conventional FFT method. Though its number of contributing averages is decreased, its output is flatter. This is also clear in Figure 14, which has an increased signal threshold of -12 dB and a further decreased number of averages. In fact, the quality of the conventional FFT method seems to degrade when signal thresholding is used. This can be attributed to the significant influence of noise therein, which increases when the number of averages decreases – even when contributing samples are chosen in a purposeful manner intended to increase the signal-to-noise ratio.

Since the conventional FFT method does not appear to benefit from signal thresholding, the remainder of Section 4.1 will only treat results generated using the *H1* method. Figure 15 shows the results of this method when several different thresholds are used.

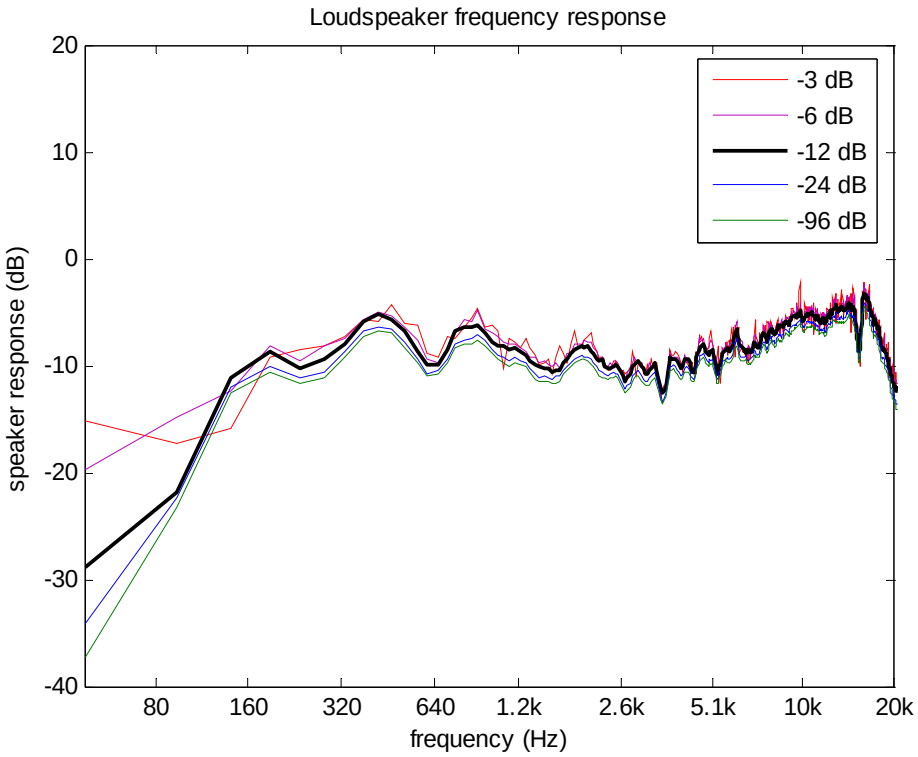

Figure 15. *H1* measurements with various signal thresholds.

Although the variation between traces in Figure 15 is minimal, extensive experimentation with multiple measurements yielded an optimal signal threshold of -12 dB. This produced the most consistent results for different values of *M*. As evidenced by Figure 15, low-frequency accuracy is actually *reduced* when the signal threshold is set too high.

4.1.3 Coherence Blanking

Following the determination of an optimum signal threshold, it was necessary to determine the maximum coherence threshold. It was found that this value varied between speakers. For this reason, it will be noted on a plot-by-plot basis. As before, the application of coherence blanking to the Alesis speaker is presented below for illustrative purposes.

The *H1* measurement of Figure 14, with a coherence threshold of 0.4, is plotted in Figure 13 against one without any kind of thresholding or blanking. Even though the processing significantly reduced the number of averages, the resultant response is noticeably flatter, and thus conforms better to the theoretical response.

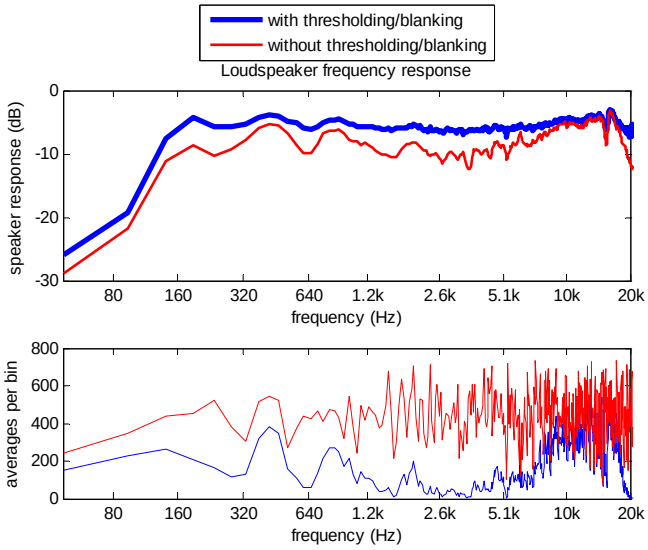

Figure 16. *H1* measurements with/without signal thresholding and coherence blanking.

The advantages of using signal thresholding and coherence blanking are clearly evidenced by Figure 16.

As a final comparison, Figure 17 shows the *H* trace of Figure 12 plotted against its fully processed, *H1* counterpart. The results present a maximum noise rejection of approximately 45 dB.

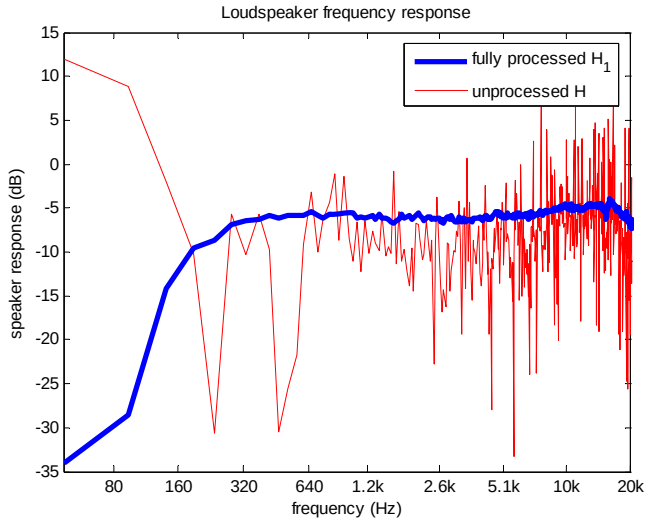

Figure 17. Processed/unprocessed loudspeaker frequency response.

As shown in Figure 17, the combined processing techniques outlined in this section serve to remove most of the non-idealities in the measurement environment. The *H1* method

and signal thresholding decreased the amount of noise present in the final measurement. Coherence blanking not only further reduced noise, but also room reflections and resonances – which would have manifested themselves as irregularities in the midrange frequencies, and resonant peaks in the lower frequencies, respectively [10]. This is most likely because reflected or resonating frequencies are generally uncorrelated with the reference signal [13].

4.1.4 Combined results

Final results for the linear characterization of each loudspeaker are shown below. The M-Audio, Alesis, and Yamaha results were generated with coherence thresholds of 0.6, 0.4, and 0.4, respectively. In each case, a measurement period of 512 samples and a 1024-point FFT were used. As a final visual enhancement, all traces were smoothed using 1/3-octave averages.

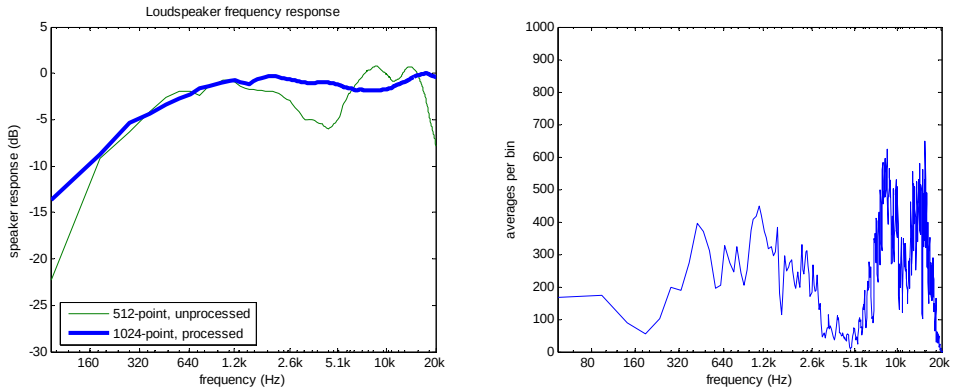

Figure 18. M-Audio loudspeaker frequency response, final.

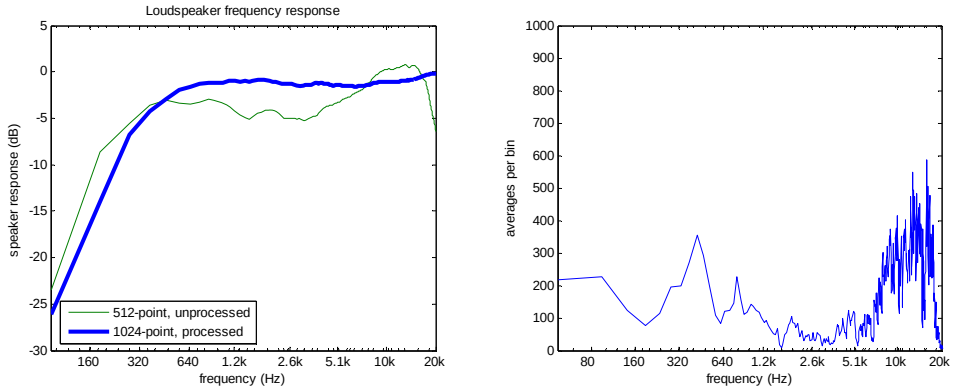

Figure 19. Alesis loudspeaker frequency response, final.

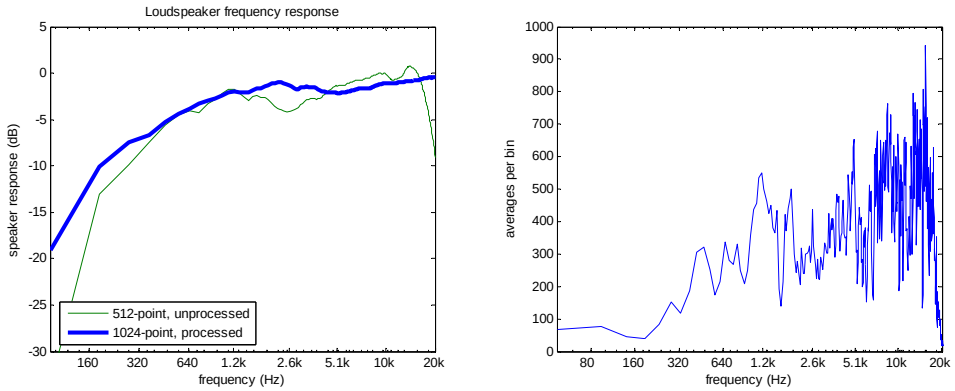

Figure 20. Yamaha loudspeaker frequency response, final.

Although detailed frequency response documentation was not available for all three speakers, each plot conforms to the expected shape. Response is approximately flat in the midrange, with sharp roll-offs at each end of the spectrum. As anticipated, the most expensive speaker – the Alesis M1 Active MkII of Figure 19 – demonstrated the flattest response and most consistent roll-offs.

The variation in the number of contributing averages per plot is most likely due to the varying signal-to-noise ratio of each measurement. An SPL (Sound Pressure Level) meter was not used to ensure consistent volume levels for all tests, so the influence of noise and reflections in each measurement is, likewise, not guaranteed to be consistent.

Also shown in Figure 18, Figure 19, and Figure 20 are unprocessed, 512-point transfer functions calculated using (3). The nonlinear results in Section 4.2 were generated using 512-point FFTs, so the green traces in these figures may be used for easy comparison. They also further demonstrate the noise-reducing benefits of signal thresholding and coherence blanking.

4.2 Nonlinear results and analysis

Plots of the two-dimensional, quadratic transfer function for the M-Audio, Alesis, and Yamaha loudspeakers can be found in Figure 21, Figure 22, and Figure 23, respectively. These plots were generated by simultaneously solving (6) for $H_1[f_k]$ and $H_2[f_i,f_j]$, via white noise excitation and the minimum mean-squared-error method given by (13). Their visual interpretation is straightforward. The values across the positive diagonal

correspond to harmonic distortion, while the remaining values correspond to inter-modulation distortions. The "trench" along the positive f_2 axis exists because no frequency interaction occurs at DC [5].

In all cases, a measurement period of 512 samples and a 512-point FFT were used.

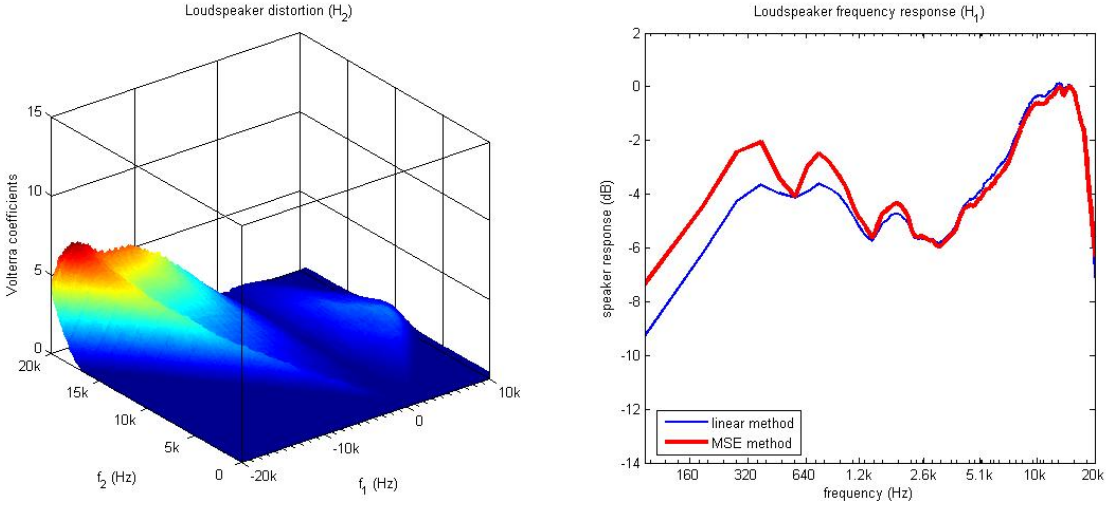

Figure 21. M-Audio quadratic and linear loudspeaker response, MMSE method.

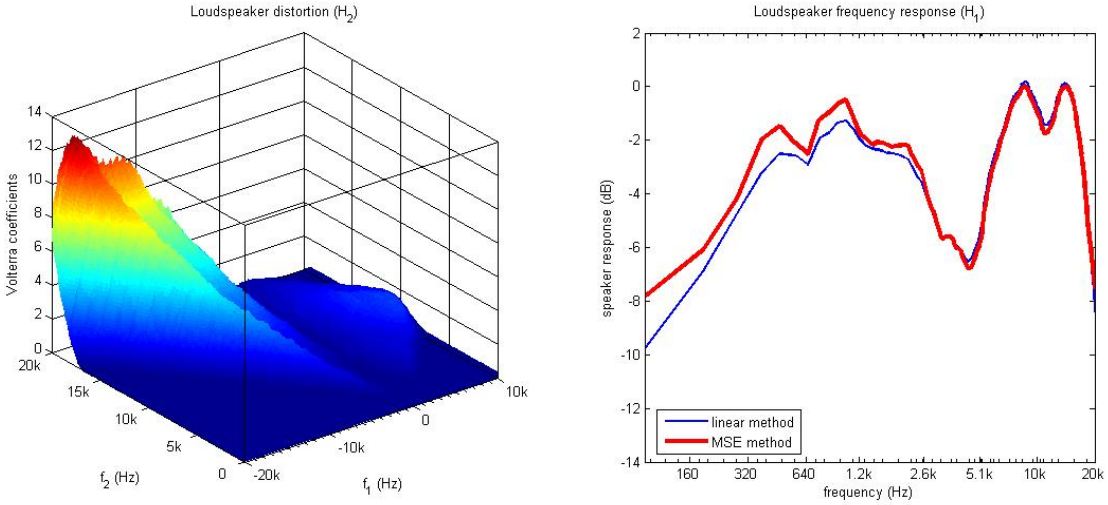

Figure 22. Alesis quadratic and linear loudspeaker response, MMSE method.

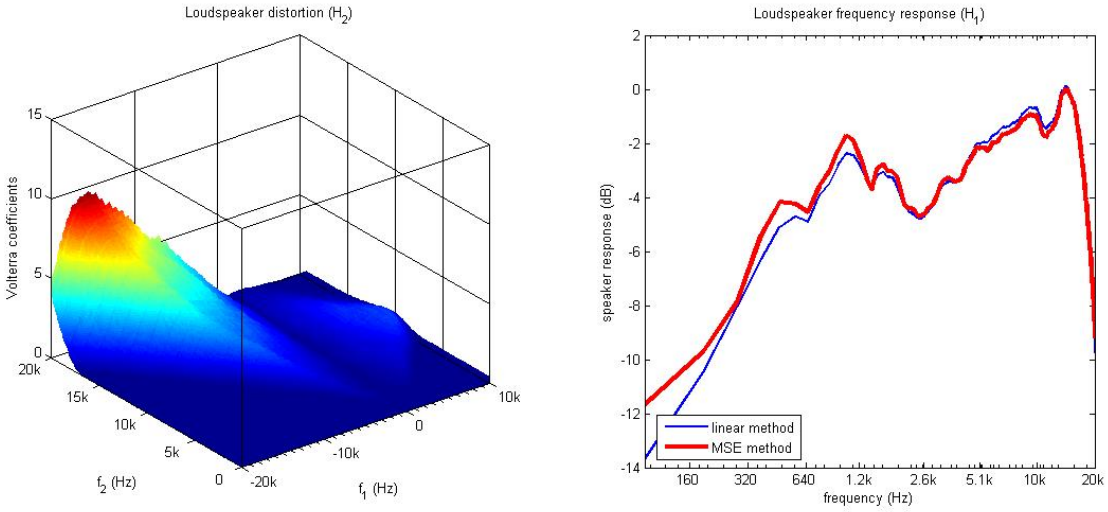

Figure 23. Yamaha quadratic and linear loudspeaker response, MMSE method.

Note that the above figures only display portions of the quadratic transfer functions which are in sum and difference regions of the two-dimensional frequency plane, as given by:

(14) $$S = \{(f_1, f_2); 0 \le f_1 \le f_2, f_1 + f_2 \le f_{N/2}\},$$

$$D = \{(f_1, f_2); -f_2 \le f_1 \le 0, -f_{N/2} \le f_1 \le 0\},$$

where S is the sum interaction region, and D is the difference interaction region, as illustrated in Figure 24.

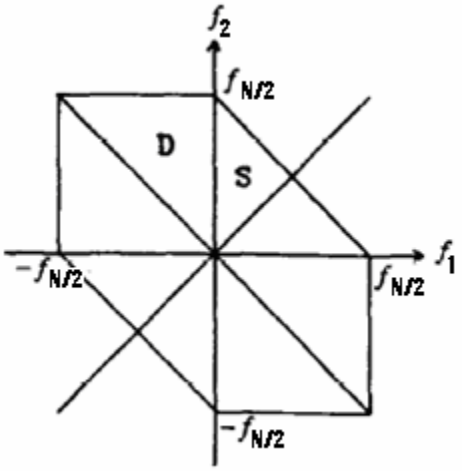

Figure 24. Two-dimensional frequency plane [16].

The quadratic transfer function $H_2[f_1,f_2]$ has unique values only in the regions S and D. Therefore, the values in all other regions can be obtained by using the symmetry property

(15) $H_2[f_1, f_2] = H_2[f_2, f_1]$, and $H_2[f_1, f_2] = H_2^*[f_1, f_2]$,

which is true for any real-valued signal [16].

In addition to quadratic transfer functions, Figure 21, Figure 22, and Figure 23 display the linear response of each speaker, as generated by both equation 3 and the MMSE method (to increase the resolution at critical frequencies, DC is not shown in these plots). The traces for the linear method were taken directly from the unprocessed traces in Figure 18, Figure 19, and Figure 20. The unprocessed traces were chosen for comparison, because no type of thresholding was applied to the spectra used to compute $H_v[k]$.

The comparison of the two frequency responses in each figure yields a strong correlation between the unprocessed *H1* and MMSE results. This is significant for three reasons: (i) it demonstrates that most loudspeaker distortion is averaged out in equation 3, (ii) in lieu of a reference, the similarity in linear results garners increased confidence in the accuracy of the MMSE algorithm, and (iii) the MMSE model most likely relegates noise to its distortion terms. If noise-reduction techniques comparable to those outlined in Section 2.3 were available for second-order systems, it is likely that the nonlinear results would agree closely with the processed plots of Figure 18, Figure 19, and Figure 20.

Figure 25, Figure 26, and Figure 27 show the harmonic distortion of each speaker plotted against its final, linear response from Section 4.1. As anticipated, all three speakers showed a generally low level of distortion, with a sharp roll-off in the higher frequencies.

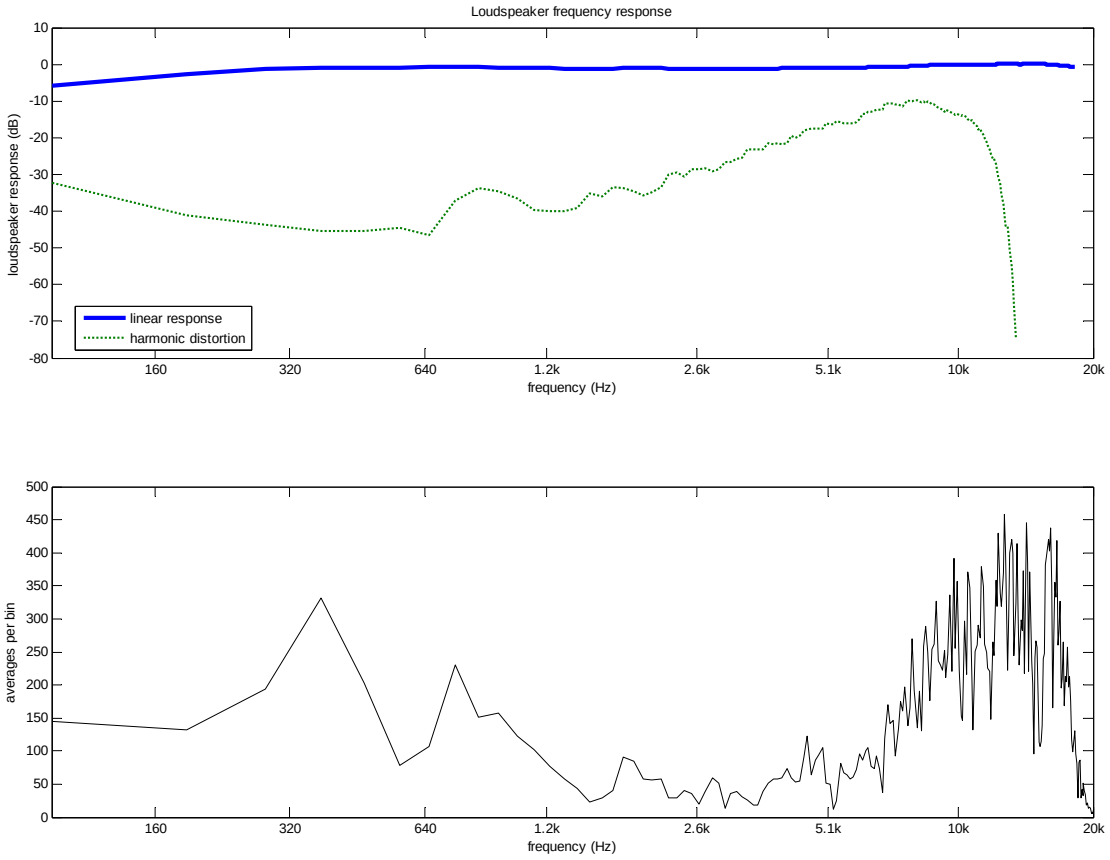

Figure 25. M-Audio loudspeaker frequency response and harmonic distortion

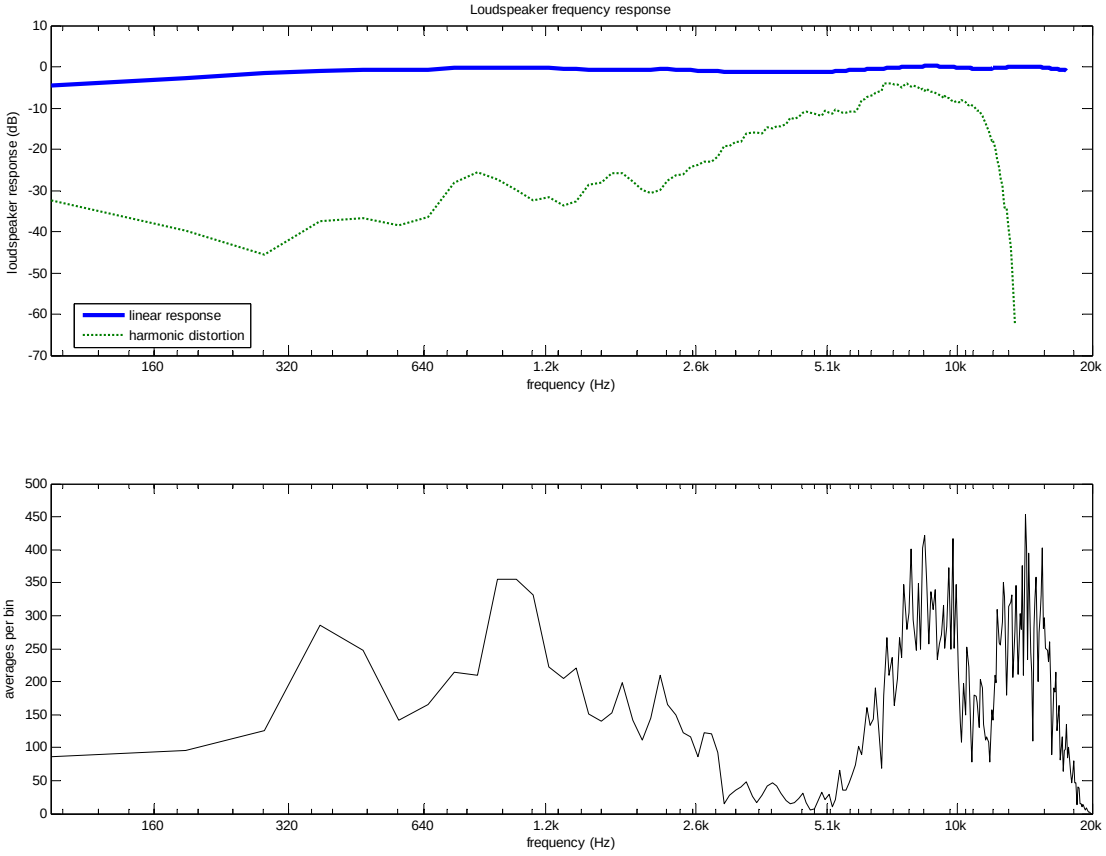

Figure 26. Alesis loudspeaker frequency response and harmonic distortion.

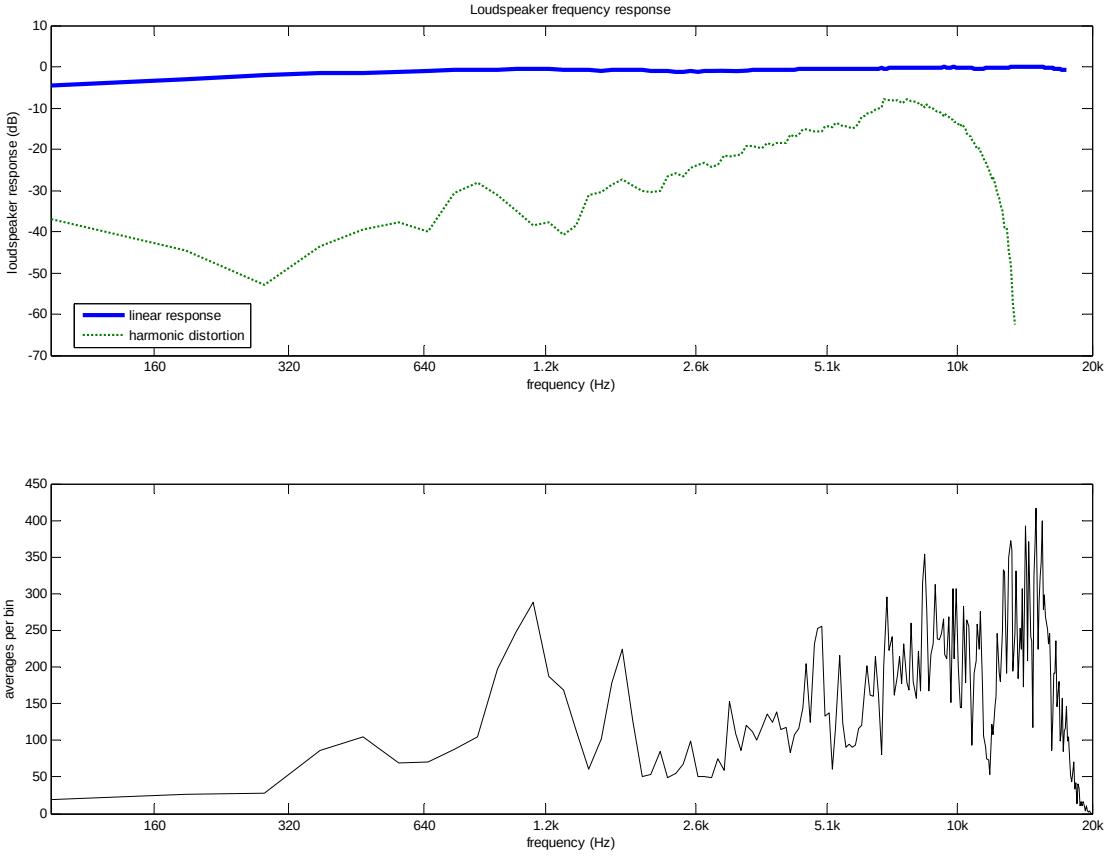

Figure 27. Yamaha loudspeaker frequency response and harmonic distortion.

The advantages of the white noise excitation techniques utilized in this report are clearly demonstrated in Figure 25, Figure 26, and Figure 27. These plots show that linear and nonlinear results may be obtained with the same basic excitation signal, without performing extensive and tedious multi-sine measurements.

4.3 Limitations of results

The noisy acoustic channel used during measurements placed limitations on some of the nonlinear results that were generated. The frequency content of this noise can be seen in Figure 28.

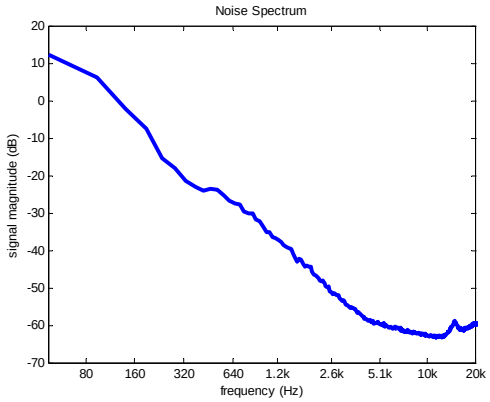

Figure 28. Frequency content of the noisy acoustic channel used during measurement.

Once the noise was acknowledged, its effect on the results had to be determined. The high concentration of energy in the first quadrant of the two-dimensional plots shown in Figure 21, Figure 22, and Figure 23 is a clear source of error. Loudspeakers cannot generate such disproportionately large amounts of energy in the involved frequency ranges. Therefore, this data is most likely a product of the noisy environment the measurements were taken in. Since the noise signal was uncorrelated with the excitation signal, the MMSE model must have relegated noise energy to its distortion terms. This explanation, however, demands further experimentation in order to be confirmed.

For the same reasons, the increase in harmonic distortion between 2.6 kHz and 10 kHz demonstrated in Figure 25, Figure 26, and Figure 27 is probably inaccurate. Further experimentation is again required to determine whether or not that is the case.

5. Conclusion

The main accomplishment of this report was outlining the characterization of dynamic loudspeakers, via white noise excitation combined with several signal processing techniques. Both linear and nonlinear characterizations were treated, and the results of the different methods were compared. This produced detailed data on the effects that distortions have on the acoustic output of loudspeakers.

The linear characterizations demonstrated that many non-idealities in the measurement environment can be removed with signal processing based on first-order coherence spectra. These processing techniques attenuated noise by as much as 45 dB. The influence of both noise and room reflections were significantly reduced, and the linear results behaved as expected.

The nonlinear characterizations demonstrated that there may be considerable harmonic and inter-modulation distortions present in loudspeaker outputs. It was also shown that distortions and/or noise have little effect on linear measurements such as frequency response. Unfortunately, the inability to perform the noise-reduction techniques utilized for the linear characterizations marginalized the nonlinear results. Techniques utilizing second-order coherence spectra, such as those described in [5], may allow one to alleviate this problem.

Further experimentation should involve the same measurements performed in a noiseless, anechoic environment. This could demonstrate the accuracy of the linear noise reduction techniques, and would also allow one to identify specific errors in the results contained herein. Such facilities were not available at the time this project was completed.

Once the findings of this report are confirmed in the manner outlined in the immediately preceding paragraph, the nonlinear characterization techniques should be expanded to account for higher-order distortions. This would allow for very accurate loudspeaker models, which could be used to simulate loudspeaker performance in software. A

particularly useful application of such models would be identifying the distortion characteristics of loudspeakers under arbitrary excitation signals, such as music or speech.

While there are multiple opportunities for further investigation, this project offered some insight into loudspeaker transfer characteristics that are often overlooked. It is clear that frequency response measurements exclude a great deal of information, and that a simple harmonic distortion percentage is not sufficient to describe the nonlinear characteristics of a speaker. The signal processing techniques described in this report offer meaningful, alternative methods of characterization.

6. References

[1] Alesis M1 Active MkII Manual. Alesis. 20 June 2006 <http://www.alesis.com/downloads/manuals/M1ActiveMKII_Manual.pdf>.

[2] Atkinson, John. "Loudspeakers: What Measurements Can Tell Us-and What They Can't Tell Us!" AES 103rd Convention. New York, August 1997.

[3] Bernstein, Julian L. Audio Systems. New York: John Wiley & Sons, Inc., 1966.

[4] Chapman, Peter J. "Programme material analysis." AES 100th Convention. Copenhagen, May 1996.

[5] Cho, Yong Soo, et al. "A Digital Technique to Estimate Second-Order Distortion Using Higher Order Coherence Spectra." IEEE Transactions on Signal Processing 40.5 (May 1992).

[6] Chavasse, P.J.A.H. and Lehman, R. "Procedures for Loudspeaker Measurements." IRE Transactions on Audio, May-June 1958.

[7] Davis, Gary, and Ralph Jones. The Sound Reinforcement Handbook. 2nd ed. Milwaukee: H. Leonard, 1989.

[8] de Vries, Ruurd, et al. "Digital compensation of nonlinear distortion in loudspeakers." IEEE International Conference on Acoustics, Speech, and Signal Processing, 1993.

[9] DPA Type 4006 Manual. DPA Microphones. 20 June 2006 <http://www.dpamicrophones.com/Images/DM02115.pdf>.

[10] Everest, F. Alton. MAster Handbook of Acoustics. 4th ed. New York: McGraw-Hill, 2001.

[11] Gabbouj, M., et al. "Modeling and real-time auralization of electrodynamic loudspeaker non-linearities." IEEE International Conference on Acoustics, Speech, and Signal Processing, 2004. Proceedings. 4 (May 2004): 81-84.

[12] Henricksen, Cliff. "Loudspeakers, Enclosures, Headphones." Handbook for Sound Engineers. 1987. Ed. Glen Ballou. 2nd ed. Boston: Butterworth-Heinemann, 1998. 497-594.

[13] Hwang, D. O., S. W. Nam, and H. W. Park. "Nonlinear echo cancellation using a correlation LMS adaptation scheme." Department of Electrical and Computer Engineering, Hanyang University, Seoul, 133-791, Korea.

[14] Bendat, J.S. and Piersol, A.G. Engineering Applications of Correlation and Spectral Analysis. New York: Wiley, 1980.

[15] Kaizer, A.J.M. "The modeling of the nonlinear response of an electrodynamic loudspeaker by a Volterra series expansion." AES 80th Convention. Montreux, Switzerland, March 1986.

[16] Kim, Kyoung Il, and Edward J. Powers. "A Digital Method of Modeling Quadratically Nonlinear Systems with a General Random Input." IEEE Transactions on Acoustics, Speech, and Signal Processing 36.11 (Nov. 1988).

[17] Klippel, Wolfgang. "Dynamic measurement and interpretation of the nonlinear parameters of electrodynamic loudspeakers." AES 88th Convention. Montreux, Switzerland, March 1990.

[18] "Loudspeaker." Wikipedia. 24 May 2006. 24 May 2006 <http://en.wikipedia.org/wiki/Loudspeaker>.

[19] "Loudspeaker measurement." Wikipedia. 25 Apr. 2006. 25 May 2006 <http://en.wikipedia.org/wiki/Loudspeaker_measurement>.

[20] "LTI system theory." Wikipedia. 16 May 2006. 24 May 2006 <http://en.wikipedia.org/wiki/LTI_system_theory>.

[21] M-Audio Studiophile SP-5B Manual. M-Audio. 20 June 2006 <http://www.m-audio.com/images/global/manuals/SP-5B_Manual.pdf>.

[22] Meyer, John. "Equalization using Voice and Music as the source." AES 76th Convention. New York, October 1984.

[23] Meyer, John. “Precision transfer function measurements using program material as the excitation signal.” AES 11th International Conference: AES Test & Measurement Conference, April 1992.

[24] Muller, Swen, and Massarani, Paulo. "Transfer-Function Measurement with Sweeps." J. Audio Eng. Soc. 49.6 (June 2001): 443-471.

[25] Randall, R. B., and B. Tech. Frequency Analysis. Bruel & Kjaer. Glostrup: K. Larson & Son, 1987.

[26] Schattschneider, Jorn, and Zolzer, Udo. “Discrete-time models for nonlinear audio systems.” 2nd COST G-6 Workshop on Digital Audio Effects. Trondheim, December 1999.

[27] Schetzen, M. The Volterra and Wiener Theories of Nonlinear Systems. New York: John Wiley & Sons, Inc., 1980.

[28] Schoulkens, and Pinkelton. "Measurement of frequency response functions in noisy environments." IEEE Transactions on Instrumentation and Measurement 39.6 (Dec. 1990).

[29] Strang, Gilbert. Linear Algebra and its Applications. 1976. 3rd ed. San Diego: Harcourt Brace Jovanovich, 1988.

[30] Tascam US-122 Manual. Tascam. 20 June 2006 <http://www.tascam.com/Products/US-122/US122_Eng.pdf>.

[31] Yamaha MSP5 Manual. Yamaha Pro Audio. 20 June 2006 <http://www2.yamaha.co.jp/manual/pdf/pa/english/speakers/MSP10E.pdf>

7. Appendix – MATLAB code

```
% --------------------------------------------------------------------
% loudspeaker_response.m
%  -this m-file prompts the user to input a reference (input) and a
%   measured (output) wave file, then computes both linear and nonlinear
%   transfer characteristics
%
%   written by Samuel Brown
%   last modified on 06/25/2006
% --------------------------------------------------------------------

% --------------------------------------------------------------------
% USER-DEFINED CONSTANTS
% --------------------------------------------------------------------

P  = 512;    % measurement period
N  = 512;    % FFT window size (>= P, P is zero-padded if N>P)
B  = P/2;   % amount of window overlap, in samples (must be less than or
          % equal to 1/2 the measurement period)
fs = 48e3;  % sampling frequency

% FFT index for 20kHz
f20kHz = ceil(20e3*(N/2+1)/(fs/2));
% calculate the frequency indices up to 20kHz
f1 = linspace(0,(f20kHz*fs/2)/(N/2+1),f20kHz);
f2 = linspace((-N/4+1)*fs/N+(fs/2-20e3)/4,(f20kHz*fs/2)/(N/2+1),...
   N/4-1+f20kHz-fix((fs/2-20e3)/4*N/fs));
% calculate 1D 1/3-octave intervals
fh = ceil((1:N/2+1).*(2^(1/6)));
fl = ceil((1:N/2+1)./(2^(1/6)));

% dB level at which signal thresholding takes place
threshold   = -12.0;
% value at which coherence thresholding takes place [0...1]
threshold_c = 0.9;

% --------------------------------------------------------------------
% FILE INPUT
% --------------------------------------------------------------------

% ask user if the workspace should be loaded from file
reply = input('load workspace from file? Y/N [Y]: ','s');
if isempty(reply)
   reply = 'Y';
end % end if

% load in the data from file, if desired
if strcmp(reply, 'Y') | strcmp(reply, 'y')
   % get the reference signal file name
   filename = input('enter file name (*.MAT) [data.mat]: ','s');
   if isempty(filename)
      filename = 'data.mat';
   end % end if
   % load the data
```

```
   disp('loading data...'); pause(.1);
   load(filename);
else % if data is not loaded from file
   % ask user if new wave files need to be loaded
   reply = input('load WAV files? Y/N [Y]: ','s');
   if isempty(reply)
      reply = 'Y';
   end % end if

   % load in the wave files, if desired
   if strcmp(reply, 'Y') | strcmp(reply, 'y')
      % get the reference signal file name
      filename = input('enter reference WAV file name (*.WAV) [reference.wav]: ','s');
      if isempty(filename)
         filename = 'reference.wav';
      end % end if
      [m d] = wavfinfo(filename); % get the file info
      disp(d);                    % display the file info
      if isempty(m) return; end;  % if there is an error (file DNE, not a
                                  % wave file), return
      % load the wave file and store the sampled data in x
      disp('loading...'); pause(.1);
      filedata = wavread(filename);
      x = filedata(:,1);  % use only the left channel, if the file is stereo

      % get the measured signal file name
      filename = input('enter measured WAV file name (*.WAV) [measured.wav]: ','s');
      if isempty(filename)
         filename = 'measured.wav';
      end % end if
      [m d] = wavfinfo(filename); % get the file info
      disp(d);                    % display the file info
      if isempty(m) return; end;  % if there is an error (file DNE, not a wave file), return
      % load the wave file and store the sampled data in y
      disp('loading...'); pause(.1);
      filedata = wavread(filename);
      y = filedata(:,1);  % use only the left channel, if the file is stereo
   end % end if

   % clear out temporary data
   clear filedata;
   clear filename;
   clear m;
   clear d;

   % ------------------------------------------------------------------
   % GENERATE FREQUENCY-DOMAIN DATA
   % ------------------------------------------------------------------

   % ask user if new spectra need to be generated
   if ~(strcmp(reply, 'Y') | strcmp(reply, 'y'))
      reply = input('re-generate spectra? Y/N [Y]: ','s');
      if isempty(reply)
         reply = 'Y';
      end % end if
   end % end if
```

```
% generate new spectra, if desired
if strcmp(reply, 'Y') | strcmp(reply, 'y')
   disp('allocating memory...'); pause(.1);
   % get the maximum number of measurement periods
   M = fix((length(y)-B)/(P-B));

   % allocate vectors to store the input and output spectra
   Gxx = zeros(M,N);
   Gyy = zeros(M,N);
   Gxy = zeros(M,N);
   X   = zeros(M,N);
   Y   = zeros(M,N);
   % allocate vectors to store the auto- and cross-correlation
   % functions
   xcorrx = zeros(M,2*P-1);
   ycorry = zeros(M,2*P-1);
   xcorry = zeros(M,2*P-1);

   disp('creating window...'); pause(.1);
   % create a hanning window for overlapping measurement periods
   w = hann(B*2);
   if(isempty(w)) window = rectwin(P);
   else           window = vertcat(w(1:B),ones(P-B*2,1),w(B+1:B*2));
   end % end if
   clear w; % clear out temp var

   disp('calculating auto- and cross-correlaton functions...'); pause(.1);
   for i = 0:M-1
      % calculate the autocorrelation of the input
      xcorrx(i+1,:) = xcorr(x(1+i*(P-B):P+i*(P-B)).*window,'coeff');
      % calculate the autocorrelation of the output
      ycorry(i+1,:) = xcorr(y(1+i*(P-B):P+i*(P-B)).*window,'coeff');
      % calculate the cross-correlation of the input and output
      xcorry(i+1,:) = xcorr(x(1+i*(P-B):P+i*(P-B)).*window,y(1+i*(P-B):P+i*(P-B)).*window,'coeff');
   end

   disp('generating auto- and cross-spectra...'); pause(.1);
   for i = 0:M-1
      % store the correlated input spectrum
      Gxx(i+1,:) = fft(xcorrx(i+1,1:P),N);
      % store the correlated output spectrum
      Gyy(i+1,:) = fft(ycorry(i+1,1:P),N);
      % store the output spectrum due to input
      Gxy(i+1,:) = fft(xcorry(i+1,1:P),N);
   end

   disp('generating fourier spectra...'); pause(.1);
   for i = 0:M-1
      % store the FFT of both x and y
      X(i+1,:) = fft(x(1+i*(P-B):P+i*(P-B)).*window,N);
      Y(i+1,:) = fft(y(1+i*(P-B):P+i*(P-B)),N);
   end % end for

   % caclulate the coherence function
   disp('generating coherence spectra...'); pause(.1);
```

```
      coherence = (abs(Gxy).^2)./(Gxx.*Gxy);

      % clear out temporary variables
      clear window;
   end % end if
end % end if

% ------------------------------------------------------------------------
% LINEAR TRANSFER CHARACTERISTIC
% ------------------------------------------------------------------------

% ask user if new linear transfer functions need to be generated
reply = input('perform linear characterization? Y/N [Y]: ','s');
if isempty(reply)
   reply = 'Y';
end % end if

% generate new liner transfer functions, if desired
if strcmp(reply, 'Y') | strcmp(reply, 'y')
   % copy over the spectral data
   disp('coying data...'); pause(.1);
   Gxx_t     = Gxx;
   Gxy_t     = Gxy;

   % find the minimum and maximum value of each bin
   disp('calculating thresholds...'); pause(.1);
   maxval = max(abs(Gxx));
   minval = min(abs(Gxx));
   % set the theshold according to the peak value
   thresh = minval+(maxval-minval).*10^(threshold/20);

   % perform signal thresholding
   disp('performing signal thresholding...'); pause(.1);
   % loop through each of the frequency bins
   for i = 1:N
      % zero out the bins that fall below the threshold
      j = find(abs(Gxx(:,i)) < thresh(i));
      Gxx_t(j,i) = 0;
      Gxy_t(j,i) = 0;
   end % end for

   % clear out temporary variables
   clear maxval;
   clear minval;
   clear thresh;

   % perform coherence blanking
   disp('performing coherence blanking...'); pause(.1);
   % zero out any bins where the coherence falls below a preset threshold
   % (typically somewhere between 0.4 and 0.8)
   j = find(abs(coherence) < threshold_c);
   Gxx_t(j) = 0;
   Gxy_t(j) = 0;

   % calculate the number of averages per bin
   disp('calculating number of averages...'); pause(.1);
```

```
   % allocate vectors to store the number of averages in each bin
   ave = zeros(1,N);
   % loop through the bin
   for i = 1:N
      % store the number of everages in this bin
      ave(i) = length(find(Gxx_t(:,i)));
   end % end for

   % calculate the average frequency-domain transfer function (in dB)
   disp('calculating average frequency-domain transfer functions...'); pause(.1);
   s = warning('off', 'MATLAB:divideByZero');
   H_fancy = abs((sum(Gxy_t)./ave)./(sum(Gxx_t)./ave));
   H_raw = abs(mean(Gxy)./mean(Gxx));
   warning(s);

   % perform 1/3-octave smoothing
   disp('performing 1/3-octave smoothing...'); pause(.1);
   % copy data
   Hsmooth     = H_fancy;
   Hsmooth_raw = H_raw;
   % loop through each of the frequency bins
   for i =2:f20kHz
      % perform averaging
      Hsmooth(i)     = mean(H_fancy(fl(i):fh(i)));
      Hsmooth_raw(i) = mean(H_raw(fl(i):fh(i)));
   end % end for
   % copy over the data
   H_fancy = Hsmooth;
   H_raw   = Hsmooth_raw;

   % clear out temporary variables
   clear Hsmooth;
   clear Hsmooth_raw;
   clear s;
   clear Gxx_t;
   clear Gxy_t;
end % end if

% ------------------------------------------------------------------------
% NONLINEAR TRANSFER CHARACTERISTIC
% ------------------------------------------------------------------------

% ask user if new nonlinear transfer functions need to be generated
reply = input('perform nonlinear characterization? Y/N [Y]: ','s');
if isempty(reply)
   reply = 'Y';
end % end if

% generate new nonliner transfer functions, if desired
if strcmp(reply, 'Y') | strcmp(reply, 'y')
   % allocate space for the transfer functions (one per frequency bin)
   disp('allocating memory...'); pause(.1);
   Hv = cell(1,N/2+1);     % Volterra coefficients
   H1 = zeros(1,N/2+1);    % one-dimensional linear transfer funtion
   H2 = zeros(N/2+1,N);    % two-dimensional nonlinear transfer function
```

```
% factor of harmonic distortion
Dh2  = zeros(1,N/2+1);

% loop through and compute the transfer characteristic, one bin at
% a time
for m = 0:N/2
   disp(sprintf('computing transfer characteristic (%d of %d)...',...
      m+1,N/2+1)); pause(.1);
   % initialize matrices and calculate indices
   if mod(m,2) % odd m
      % calculate the number of coefficients
      n = N/2+2-(m+1)/2;
      % initialize matrices
      R  = zeros(n,n);
      S  = zeros(n,1);
      Xv = zeros(n,1);
      % calculate the indices of the sum and difference terms
      f_i = (m+(1:2:N-m))/2;  % calculate the i indices
      f_j = (m-(1:2:N-m))/2;  % calculate the j indices
   else       % even m
      % calculate the number of coefficients
      n = N/2+2-m/2;
      % initialize matrices
      R = zeros(n,n);
      S = zeros(n,1);
      Xv = zeros(n,1);
      % calculate the indices of the sum and difference terms
      f_i = m/2+(0:1:(N-m)/2);  % calculate the i indices
      f_j = m/2-(0:1:(N-m)/2);  % calculate the j indices
   end % end if

   % wrap indices around
   w = find(f_j < 0);
   f_j(w) = f_j(w)+N;

   % loop through all the measurement periods
   for k = 1:M
      % construct the X matrix
      Xv(:) = horzcat(X(k,m+1), X(k,f_i+1).*X(k,f_j+1));
      % update the averages on the RHS of the minnimum mean square eqn
      R = R+conj(Xv)*Xv.'/M;
      S = S+conj(Xv)*Y(k,m+1)/M;
   end % end for

   % compute the coefficients for this frequency bin (m)
   Hv{m+1} = inv(R)*S;
   % update the linear transfer function
   H1(m+1) = abs(Hv{m+1}(1));
end %end for

% clear out temporary variables
clear Xv;
clear k;
clear R;
clear S;
```

```
% construct the 3D, nonlinear transfer function from the Volterra
% coefficients
disp('constructing 3D, nonlinear transfer function...'); pause(.1);
for m = 0:N/2
   % initialize matrices and calculate indices
   if mod(m,2) % odd m
      % calculate the indices of the sum and difference terms
      f_i = (m+(1:2:N-m))/2;  % calculate the i indices
      f_j = (m-(1:2:N-m))/2;  % calculate the j indices
      % wrap sum and difference indices around
      w = find(f_j < 0);
      f_j(w) = f_j(w)+N;
      % construct the 2D transfer function for this bin
      for index = 2:N/2+2-(m+1)/2
         H2(f_i(index-1)+1,f_j(index-1)+1) = abs(0.5*Hv{m+1}(index));
      end % end for
   else       % even m
      % calculate the indices of the sum and difference terms
      f_i = m/2+(0:1:(N-m)/2);  % calculate the i indices
      f_j = m/2-(0:1:(N-m)/2);  % calculate the j indices
      % wrap sum and difference indices around
      w = find(f_j < 0);
      f_j(w) = f_j(w)+N;
      % construct the 2D transfer function for this bin
      H2(m/2+1,m/2+1) = abs(Hv{m+1}(2)); % no factor of 2 for the first coeff
      for index = 3:N/2+2-m/2
         H2(f_i(index-1)+1,f_j(index-1)+1) = abs(0.5*Hv{m+1}(index));
      end % end for
   end % end if
end % end for

% perform 1/3-octave smoothing
disp('performing 1/3-octave smoothing...'); pause(.1);
% calculate 2D 1/3-octave intervals
subt = (1:N/2+1)-ceil((1:N/2+1)./(2^(1/6)));
addt = ceil((1:N/2+1).*(2^(1/6)))-(1:N/2+1);
fl2 = [(1:N/2+1)-subt (N/2+2:N)-addt(end-2:-1:1)];
fh2 = [(1:N/2+1)+addt (N/2+2:N)+subt(end-2:-1:1)];
w = find(fl2(N/2+2:N) < N/2+2);
fl2(N/2+1+w) = N/2+2;
w = find(fh2(1:N/2+1) > N/2+1);
fh2(w) = N/2+1;

% copy data
H1smooth  = H1;
H2smooth  = H2;
Dh2smooth = Dh2;
% loop through each of the frequency bins
for i =2:f20kHz
   % perform averaging
   H1smooth(i)  = mean(H1(fl(i):fh(i)));
   Dh2smooth(i) = mean(Dh2(fl(i):fh(i)));
   for j = 1:N
      count = 0;
      for m = fl(i):fh(i)
         for n = fl2(j):fh2(j)
```

```
            H2smooth(i,j) = H2smooth(i,j)+H2(m,n);
            count = count+1;
          end % end for
        end % end for
      end % end for
      H2smooth(i,j) = H2smooth(i,j)/count;
   end % end for
   % copy over the data
   H1  = H1smooth;
   H2  = H2smooth;
   Dh2 = Dh2smooth;

   % construct the factor of harmonic distortion
   for m = 0:N/2
      Dh2(m+1) = abs(0.5*H2(m+1,m+1))/H_fancy(m+1);
   end % end for

   % clear out temporary variables
   clear fh;
   clear fl;
   clear fh2;
   clear fl2;
   clear count;
   clear H1smooth;
   clear H2smooth;
   clear Dh2smooth;
   clear f_i;
   clear f_j;
   clear m;
   clear n;
   clear index;
   clear Hv;
   clear w;
end % end if

% ------------------------------------------------------------------------
% SAVE RESULTS TO FILE
% ------------------------------------------------------------------------

% ask user if they want to save the workspace
reply = input('write workspace to file? Y/N [Y]: ','s');
if isempty(reply)
   reply = 'Y';
end % end if

% load in the wave files, if desired
if strcmp(reply, 'Y') | strcmp(reply, 'y')
   % get the reference signal file name
   filename = input('enter file name (*.MAT) [data.mat]: ','s');
   if isempty(filename)
      filename = 'data.mat';
   end % end if
   % clear out remaining temp vars
   clear reply;
   clear h;
   clear i;
```

```
   clear j;
   % save the data to file
   disp('saving data...'); pause(.1);
   save(filename);
else
   % just clear the vars
   clear reply;
   clear h;
   clear i;
   clear j;
end % end if

% ------------------------------------------------------------------------
% LINEAR RESULTS OUTPUT
% ------------------------------------------------------------------------

% get the scaling factor between the MSE and linear results
sfact  = mean((H_raw(1:f20kHz)./max(H_raw(1:f20kHz)))./(abs(H1(1:f20kHz))./max(H1(1:f20kHz))));
sfact2 =
mean((H_fancy(1:f20kHz)./max(H_fancy(1:f20kHz)))./(abs(H1(1:f20kHz))./max(H1(1:f20kHz))));

% output the results
disp('displaying linear results...'); pause(.1);
figure(1);

% display the frequency response
subplot(2,1,1);
semilogx(f1,20*log10(H_fancy(1:f20kHz)./sfact./max(H_fancy(1:f20kHz))),...
   'LineWidth',3,'Color','blue');
hold all;
% display the harmonic distortion
semilogx(f1,20*log10(abs(Dh2(1:f20kHz)).*max(H_fancy(1:f20kHz))),...
   'LineWidth',1.5,'LineStyle',':','Color',[0 0.5 0]);
% set the x and y limits
set(gca,'XLim',[0 20e3]);
% set the x ticks
set(gca,'XTick',[20 40 80 160 320 640 1.2e3 2.6e3 5.1e3 10e3 20e3],...
   'XMinorTick','off');
set(gca,'XTickLabel','20|40|80|160|320|640|1.2k|2.6k|5.1k|10k|20k');
% insert a legend
legend('linear response','harmonic distortion');
% label the axes
xlabel('frequency (Hz)');
ylabel('loudspeaker response (dB)');
% label the plot
title('Loudspeaker frequency response');

% display the number of averages
subplot(2,1,2);
semilogx(f1,ave(1:f20kHz),'LineWidth',1,'Color','black');
% set the x and y limits
set(gca,'XLim',[0 20e3]);
% set the x ticks
set(gca,'XTick',[20 40 80 160 320 640 1.2e3 2.6e3 5.1e3 10e3 20e3],...
   'xminortick','off');
set(gca,'XTickLabel','20|40|80|160|320|640|1.2k|2.6k|5.1k|10k|20k');
```

```
% label the axes
xlabel('frequency (Hz)');
ylabel('averages per bin');

% clear out temporary variables
clear ax;
clear h1;
clear h2;

% ----------------------------------------------------------------------
% NONLINEAR RESULTS OUTPUT
% ----------------------------------------------------------------------

% output the results
disp('displaying nonlinear results...'); pause(.1);
figure(2);

% shift the transfer function so that DC is centered
H2 = fftshift(H2,2);

%plot the 3D portion
subplot(1,2,1);

% plot the 3D, nonlinear transfer function
s = warning('off', 'MATLAB:log:logOfZero');
surf(gca,f2(end:-1:1),f1,(abs(H2(1:f20kHz,N/16+1:length(f2)+N/16))),...
   'EdgeColor','flat','FaceLighting','phong');
warning(s);

% set up the grid
set(gca,'GridLineStyle','-','MinorGridLineStyle','none');
% set the axes limits
set(gca,'YLim',[0 20e3],'XLim',[-10e3 20e3]);
% set the axes direction
set(gca,'XDir','reverse');
% set the tick marks
set(gca,'YTick',[0 5e3 10e3 15e3 20e3],...
   'YMinorTick','off');
set(gca,'YTickLabel','0|5k|10k|15k|20k');
set(gca,'XTick',[-10e3 0 10e3 20e3],'XMinorTick','on');
set(gca,'XTickLabel','10k|0|-10k|-20k');
% turn the box on
set(gca,'box','on');
% label the axes
xlabel('f_{1} (Hz)');
ylabel('f_{2} (Hz)');
zlabel('Volterra coefficients');
% label the plot
title('Loudspeaker distortion');

% shift the transfer function back to its original state
H2 = fftshift(H2,2);

%plot the 2D portion
subplot(1,2,2);
```

```
% plot the 2D, nonlinear transfer function
semilogx(f1,20*log10(H_raw(1:f20kHz)./sfact./max(H_raw(1:f20kHz))),...
   'LineWidth',1.5,'Color','blue');
hold all;
semilogx(f1,20*log10(H1(1:f20kHz)./max(H1(1:f20kHz))),'LineWidth',3,...
   'Color','red');
hold off;
% insert a legend
legend('linear method','MSE method');
% set the x limits
xlim([0 20e3]);
% set the x ticks
set(gca,'XTick',[20 40 80 160 320 640 1.2e3 2.6e3 5.1e3 10e3 20e3],...
   'XMinorTick','off');
set(gca,'XTickLabel','20|40|80|160|320|640|1.2k|2.6k|5.1k|10k|20k');
% label the axes
xlabel('frequency (Hz)');
ylabel('speaker response (dB)');
% label the plot
title('Loudspeaker frequency response');

% clear out temporary variables
clear f20kHz;
clear f1;
clear f2;

% fin!
disp('done.');
```