A Tutorial on Non-Linear Distortion or Waveshaping Synthesis
Author(s): C. Roads
Source: *Computer Music Journal*, Vol. 3, No. 2 (Jun., 1979), pp. 29-34
Published by: The MIT Press
Stable URL: https://www.jstor.org/stable/3680281

A Tutorial on Non-linear Distortion or Waveshaping Synthesis

C. Roads

Introduction

In recent issues of *Computer Music Journal* there have been a number of references to a technique of sound synthesis which is variously termed "non-linear distortion," "non-linear processing," or "waveshaping". This technique has begun to surface as an economical and flexible method of sound synthesis, capable of generating sounds as rich as those produced through computationally costly additive synthesis techniques. In addition, the technique offers a unified theoretical framework for the understanding of several other nonlinear synthesis methods, notably frequency modulation and discrete summation synthesis.

The terms "non-linear distortion" and "waveshaping" have yet to be explained in these pages, and with the recent public availability of three major technical papers on the subject (Arfib : 1978; Beauchamp : 1979; and Le Brun : 1979) the need for a tutorial here is made more timely. Although a precise and comprehensive formulation of the results of the various forms of non-linear distortion or waveshaping involve a great deal of calculation, the concepts underlying the technique can be presented in a straightforward, largely non-mathematical form. This brief tutorial is intended particularly as an orientation and a pointer to the more technical papers cited above. Hopefully, a musician unfamiliar with signal processing tools save for electronic music equipment and a MUSIC V-like (Mathews : 1969) graphic "patching" language will be able to understand the basic concepts of this powerful technique through this exposition. Of course, as Le Brun (1979) points out, research into aspects of non-linear distortion or waveshaping is continuing, and hopefully, future *Computer Music Journal* articles will cover these developments in a timely manner.

Historical Note

Historically, the technique was "in the air" a few years ago, and appears to have been conceived by several people in parallel. According to Arfib (1978), Risset used the technique to simulate clarinet tones in 1969. Indeed, example number 150 of Risset's *Catalog of Computer Synthesized Sounds* is a description of an instrument in which a sine wave is "submitted to a non-linear transfer function" and the amplitude control of the sine wave "determines the amount of distortion performed on the sine wave" (Risset: 1969). In 1970, R. W. Schaefer presented the technique in an analog implementation—a form in which the technique is still viable, if somewhat less easy to control precisely, and introduced with Suen (1970) the foundation of the mathematical techniques which can be used to describe the spectra produced by non-linear waveshaping. Hutchins (1976) is a clear and concise exposition of the theory of non-linear transfer functions in an analog context. The present formulation of the technique is an elaboration upon these ideas worked out independently by Arfib and Le Brun.

A Description of the Basic Technique

The technique starts classically with a sine or cosine wave, the amplitude of which is controlled by an envelope generator, shown in a MUSIC V-like graphic notation in Figure 1 below. This sine wave is fed into a "non-linear processing" device. The device is called non-linear because whatever is presented to it as input is "distorted" or transformed by the time that it is output. Hence the derivation of the terms "non-linear distortion."

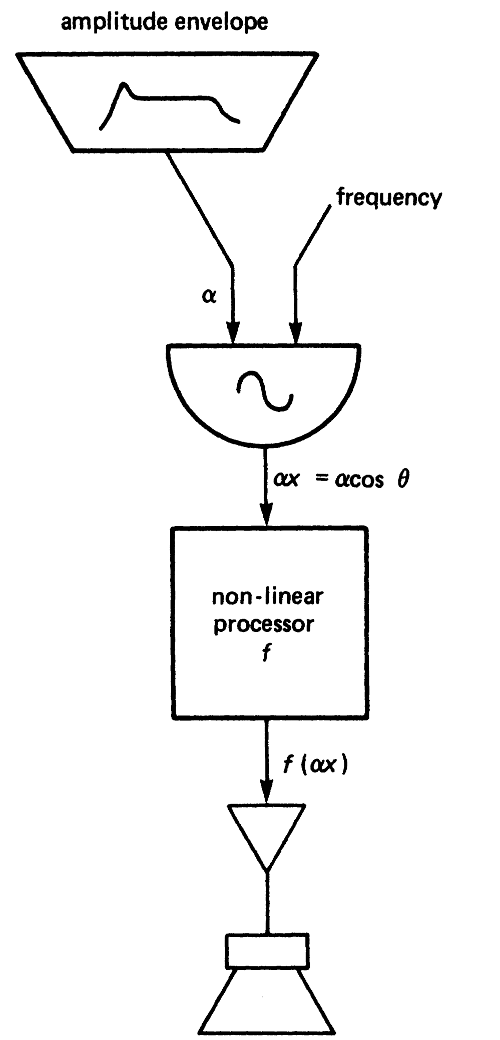


Figure 1.

The non-linear processor (or "waveshaper") may be likened to a distorting amplifier—various frequencies fed into such an amplifier will be treated unequally; some such frequency components will be suppressed, others will be exaggerated, while a whole new set of spectral components not present in the input signal may be introduced by the non-linear processor. These new spectral components are called *modulation products*. By turning the volume knob of most any amplifier to maximum, one notices how by saturating the transistors of the amplifier the input signal is subjected to "overmodulation distortion" or "clipping," which in its grossest form will convert a sine wave sent into such an amplifier into a square wave, and for a complex signal will introduce so many modulation products that the input signal itself is blotted out by distortion. When reproducing or recording music, such non-linear processing is highly undesirable, since distortion of this type can destroy the fidelity of the recording as compared to the original sound source. On the other hand, in an electronic music studio, distortion can be a highly useful technique in analog electronic music synthesis for "enriching" a sound by using preamplifiers with various distortion characteristics as non-linear processors. For example, tube and transistor/op amp amplifiers have entirely different distortion characteristics.

This naturally leads to the question of how one describes the distortion characteristics of various non-linear processors. One useful method is that of describing the *transfer function* of the non-linear processor.

The Transfer Function

In papers on the technique of non-linear distortion or waveshaping, the transfer function or "shaping function" in (Le Brun : 1979) is usually represented as a quasi-diagonal line from the lower left corner of a square to the upper right corner. In order to understand this representation, one might refer back to the amplifier analogy. The transfer function of an ideal amplifier is a straight line; hence, such an amplifier is said to have a "linear" response. The input signal is represented on a normalized scale of values between -1 and $+1$ on the abscissa (x-axis); the corresponding output signal is represented on the ordinate (y-axis). (See Figure 2)

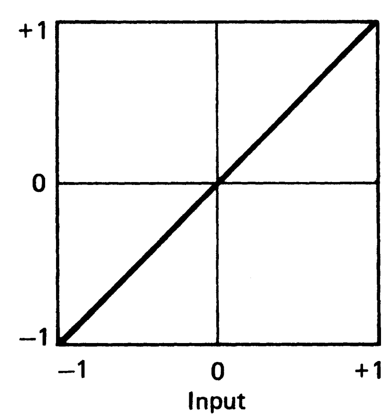


Figure 2. Transfer or Shaping Function F

A good way to acquaint oneself with the effects of non-linear distortion or waveshaping is to gain an intuitive visual understanding of the transfer function. Perhaps an easy way to gain this intuitive knowledge would be to play with the image of a transfer function grid on a visual display device such as a crt terminal with graphics capability. (See some examples in Figures 3, 4, 5, and 6.) Given a computer music

Figure 3.

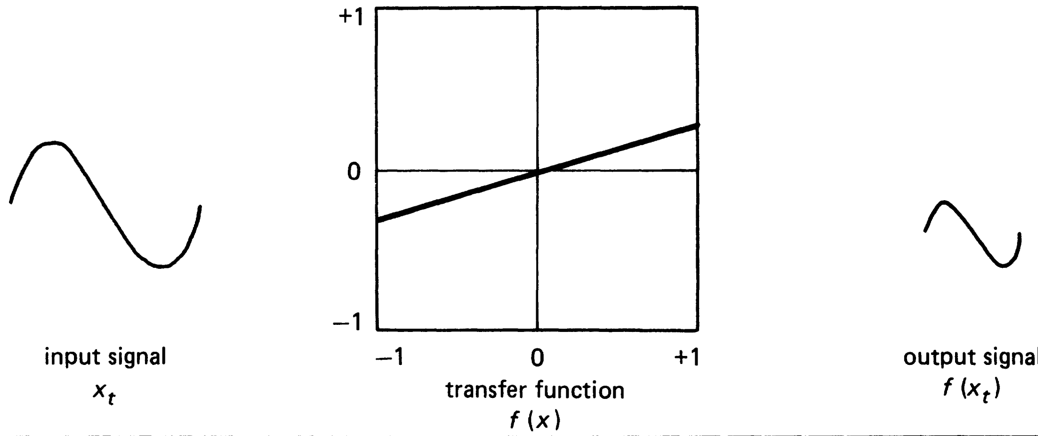


Figure 4.

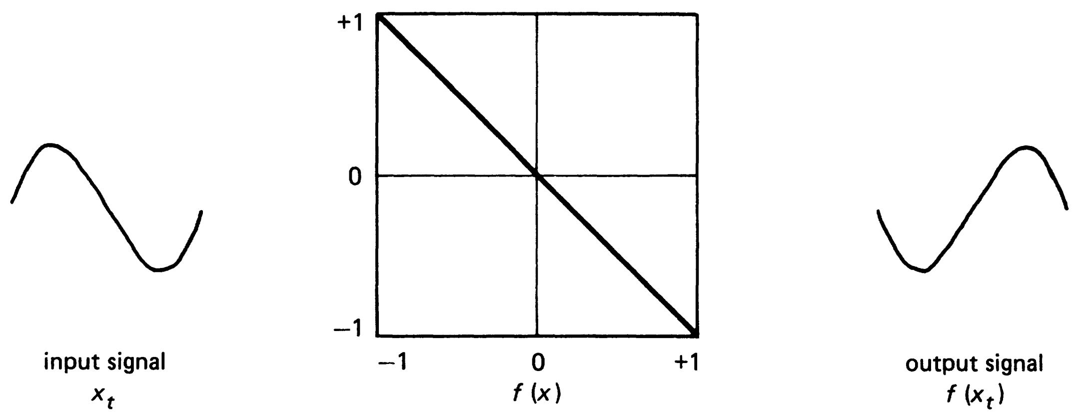


Figure 5.

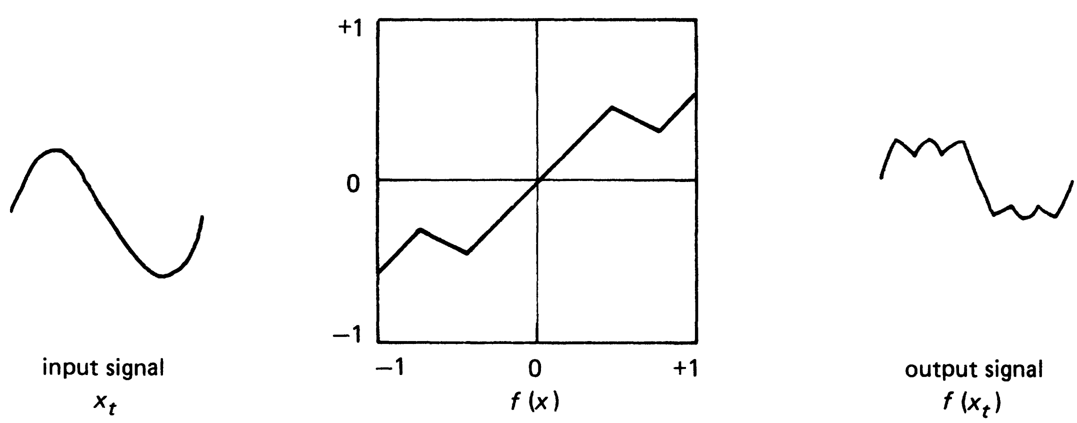
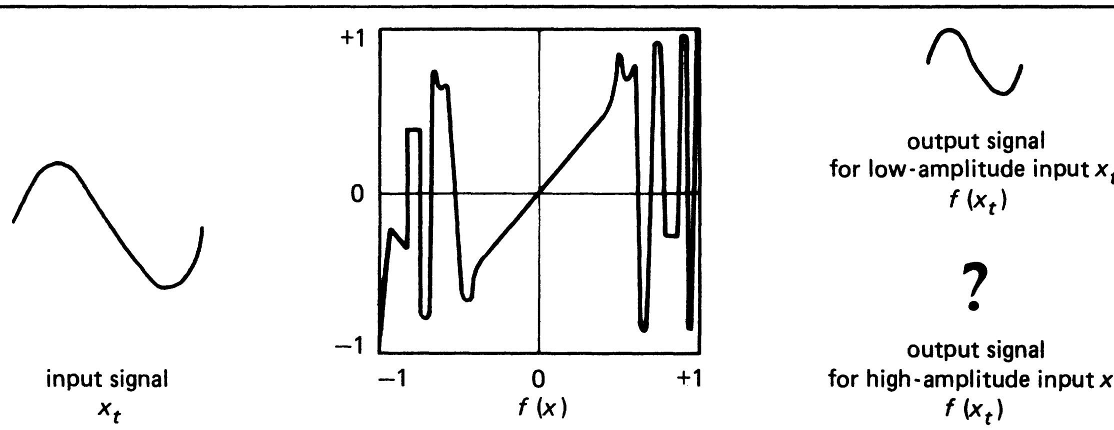


Figure 6.



system with some interactive sound synthesis capability, one could use some kind of a stylus to draw various transfer functions and then send various sounds through them, playing with their effects in real-time.

The effect of a few simple transfer functions can be described in general terms. If one had drawn the straight-line shown in Figure 3, the effect would be, for any signal input, a signal with identical waveform shape, but at a reduced amplitude. Using the line shown in Figure 4, one sees that for a positive value input to the transfer function (x-axis) a negative value is obtained at the output (y-axis). Thus, applying this transfer or shaping function results in phase inversion (or to be consistent with our terminology—phase distortion). Any kind of a bump or irregularity in the transfer function will result in a corresponding distortion applied to the output waveform. See Figure 5 for an example. As a final example, of this type, the amplitude-sensitive nature of the technique can be demonstrated. In Figure 6, one sees a transfer function characterized by a straight-line in the middle (low-amplitude) range of the grid. Such a function will pass a low-amplitude signal through with no distortion. However, when the amplitude of the input signal increases, the extreme ends of the transfer function (acting on the peak and trough of a high-amplitude sine wave) are subjected to a rather complicated form of distortion! It is easy to see that the most general behaviour of a signal passed through such a transfer function is not unlike that of most conventional musical instruments, in that by playing "harder" (*i.e.*, blowing harder, bowing with more pressure, striking more sharply, *etc.*) the spectrum of the musical output is enriched. Thus, by passing a signal whose overall amplitude varies over time, one obtains a corresponding *time-varying spectrum* output from the non-linear processor or waveshaper. This is an important feature of the technique; given a single transfer function, a variety of output waveforms may be obtained simply by varying the amplitude (and/or DC offset) of the input signal in order to apply various regions of the transfer function.

Another convenient property of transfer functions is that any of these functions drawn to be symmetric about the x-axis (the definition of an odd function) will generate only odd harmonics (See Figure 7 below).

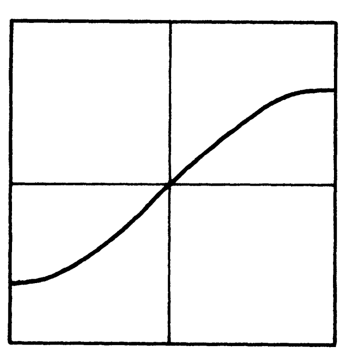


Figure 7. An odd function.

A transfer function drawn symmetric about the y-axis (an even function) will generate only even harmonics of a cosine wave input. (See Figure 8)

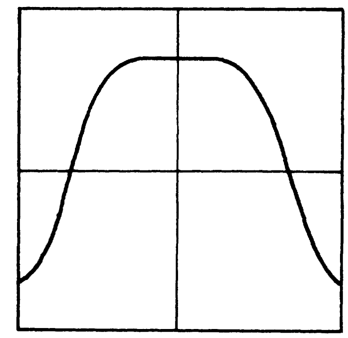


Figure 8. An even function.

A More Analytic Approach to Transfer Functions

Drawing transfer or shaping functions intuitively by hand is one way to become acquainted with their effects. However, beyond a certain degree of visual complexity, it may become difficult to predict the effects of a given transfer function. Thus, for more predictable results, a more analytic approach is needed.

A well-developed branch of mathematics is the theory of approximation of functions. This theory provides a number of tools for constructing arbitrary curves (like transfer functions) out of elementary functions such as polynomials, trigonometric functions (Moore: 1978a), and other orthogonal functions. The foundations of approximation theory were laid in the nineteenth century. Weierstrass proved theoretically that any continuous function could be approximated by an algebraic polynomial with any degree of accuracy. Bernoulli proved the theoretical possibility of representing an arbitrary function by trigonometric polynomials with any degree of accuracy. There are several advantages to constructing curves out of these kinds of functions. The main advantage is this: through various mathematical identities (*i.e.,* ways of relating one function to another) some formulae have been derived which, given a mathematical description of an input signal and another mathematical description of a transfer function, make it possible to exactly predict the output spectrum generated by the process of non-linear distortion or waveshaping. As Arfib (1978) points out, the first benefit of using a limited-degree polynomial for the transfer function is that the output will be band-limited. This of course makes it possible to predict whether foldover will occur, and thus, foldover can be avoided. (Foldover (or aliasing) occurs in digital systems when frequencies above half the sampling rate reflect into the lower range of the spectrum.)

Arfib (1978) and Le Brun (1979) used a particular family of polynomials, the *Chebychev polynomials of the first kind* as a tool for specifying shaping or transfer functions. Conveniently, they take on values in the range of the transfer functions $[-1, +1]$. For Chebychev polynomials, the following very useful identity holds, making it possible to plug in various formulae for T_k to obtain the kth harmonic:

$$T_k(\cos\theta) = \cos k\theta$$

Equations for T_0 through T_8 are given below:

$$\begin{aligned}
T_0 &= 1 \\
T_1 &= x \\
T_2 &= 2x^2 - 1 \\
T_3 &= 4x^3 - 3x \qquad\qquad \text{(where } x = \cos\theta\text{)} \\
T_4 &= 8x^4 - 8x^2 + 1 \\
T_5 &= 16x^5 - 20x^3 + 5x \\
T_6 &= 32x^6 - 48x^4 + 18x^2 - 1 \\
T_7 &= 64x^7 - 112x^5 + 56x^3 - 7x \\
T_8 &= 128x^8 - 256x^6 + 160x^4 - 32x^2 + 1
\end{aligned}$$

(For more Chebychev equations, look in any handbook of mathematical functions.)

In order to obtain a wide range of harmonics, say, from the ith to the nth harmonics, one simply forms the weighted sum of a set of Chebychev polynomials T_i to T_n according to the following formula:

$$f(x) = \frac{h_o}{2} + \sum_{k=i}^{n} h_k T_k(x)$$

where $f(x)$ is the transfer or shaping function,
$\frac{h_o}{2}$ is a conventional offset
$\sum$ signifies the summing operation for i to n,
h_k is the weight (amplitude) of the kth harmonic,
and $T_k(x)$ is the Chebychev polynomial which, when used as a transfer function, will generate the kth harmonic.

A major musical implication of this formulation is that it is possible to control even and odd harmonics, upper and lower harmonics, and closely-spaced and widely-dispersed harmonics all independently, without affecting other individual partials. Both Arfib (1978) and Le Brun (1979) give details on constructing these polynomial transfer functions. In particular, they show how the coefficients of the Chebychev polynomials are related to the amplitudes of the harmonics. Arfib then discusses an implementation in the context of the MUSIC V sound synthesis program, while Le Brun provides program listings written in an ALGOL variant.

Extensions to the Basic Technique

What has been described so far is a simple instrument which accepts a cosine wave as input, and passes it through a transfer or shaping function to produce a steady-state sound whose harmonic spectrum can be determined in advance. Many extensions to this basic configuration are possible. Some of them are discussed informally below.

Obtaining Dynamic Spectra

In the basic configuration, it was assumed that the cosine wave which was sent through the transfer function was of constant amplitude, *i.e.*, α of Figure 1 was equal to one. Computation of a steady-state spectrum relies on this steady amplitude. However, if a curved envelope function is used for α, one obtains *dynamic spectra* as the output of the non-linear processor. In general, one can expect an overall spectral evolution similar to that of the well-known fm technique, in that the greater the amplitude α (functioning similarly to the "index" in the fm theory) of the input signal, the more harmonics will be generated. Intuitively, this makes sense because as one increases the amplitude of the input, more of the bends and curves of the transfer function are being applied to the input, producing ever more complicated distortion and hence, a broader band of output sound. Suen (1970) gives a mathematical derivation of the dynamic behaviour of a non-linear processor, and points out how the evolution of individual harmonics is *not* a linear function of the input signal's amplitude α.

Obtaining Inharmonic Spectra

In the basic configuration, the output of the non-linear processor is always a spectrum with components whose frequencies were simple integer ratios (harmonics). One way to obtain more complex spectra has been suggested by both Arfib and Le Brun. This process involves amplitude modulating the output of the non-linear processor $f_{\alpha x}$ with some other frequency f_{am}. The effect of amplitude modulation is to produce a pair of sum and difference frequencies or sidebands symmetric about the frequency f_{am}. If $f_{\alpha x}/f_{am}$ is not a simple integer ratio, inharmonic sum and difference frequencies will be generated.

Normalization or "Post-Compensation" of Amplitude

In generating dynamic spectra, the input envelope or index α is used as a control variable to determine the amount of distortion (and hence the bandwidth) of the sound. An inflexible side-effect of using α in this way is that it also lowers or raises the overall amplitude coming out of the non-linear processor. What if one chooses to sweep over the entire transfer function (to obtain constantly evolving spectra) while maintaining a constant output amplitude? Some kind of "post-compensation" (*i.e., after* the non-linear processor) or amplitude normalization is clearly required. The function of this normalization is simply to be able to control the output amplitude independent of the index α. The problem is perhaps not as simple as it would seem at first glance, and various scaling functions may be considered (*cf.* Le Brun: 1979).

Controlling Phase

Controlling the phase of various components in a spectrum can sometimes be useful. One way of controlling the phase of the cosine components x_k produced by waveshaping is to generate an additional signal y which is a sine wave of an amplitude determined by the phase one wishes to obtain. This sine wave y may then be passed through a separate transfer or shaping function U_{n-1} to produce a set of spectral components y_k. Each component y_k corresponds to a component x_k in the original signal, since the transfer functions T and U are related. Indeed, U_{n-1} is a *Chebychev polynomial of the second kind* obtained by differentiating $T_n(x)$ with respect to x and dividing by n. Summing the two waveforms x_k and y_k effectively shifts the cosine wave to the desired phase.

Other Extensions

For even more flexible (and complicated) types of spectra and spectral evolution, a number of additional extensions can be mentioned. First, one may choose to add a dynamically-variable filter onto the output of the non-linear processor, as Beauchamp (1979) has done. There are a number of reasons one might want to do this; suffice it to say here that a dynamic filter can add another degree of control (and thus, variation) to the dynamic evolution of the spectra produced by a non-linear processor. (See Beauchamp: 1979 for more details.)

Another extension to the basic configuration suggested by Arfib (1978) is to multiply *two* distorted signals x_1 and x_2 by a sine wave modulating at frequency f_{am}. When x_1, x_2, and f_{am} are carefully chosen, rich inharmonic spectra and formant structures may be generated.

A last important extension is to substitute a complex signal for x in Figure 1, such as a rich electronic sound or a concrete sound like a voice, to obtain a "waveshaped" sound, using various transfer or shaping functions.

Conclusions

The technique of non-linear distortion or waveshaping appears to be a flexible, computationally-efficient method of generating a rich and varied collection of sounds. It should be quite feasible to implement the technique in software using the newer sixteen-bit microprocessors for real-time or near real-time synthesis of sound. Several digital synthesizer designs already incorporate means for realizing the technique into their hardware. It is reasonable to expect that non-linear distortion or waveshaping will become a widely-used technique, both for synthesis and processing of musical sound.

Acknowledgements

I would like to thank my friend and teacher Leonard Cottrell of Berkeley, California for his counsel on the subject of this article, and for his long-standing support. Thanks also to John Snell for his careful reading and comments on the draft of this manuscript.

Bibliography

Arfib, D. (1978) "Digital Synthesis of Complex Spectra by means of Multiplication of Non-linear Distorted Sine Waves," *AES Preprint No. 1319 (C-2)*, also in *Proceedings of the 1978 International Computer Music Conference*, Roads (Comp.), available from Northwestern University Press, Evanston.

Beauchamp, J. (1979) "Brass Tone Synthesis by Spectrum Evolution Matching with Nonlinear Functions," *Computer Music Journal*, Vol. 3, No. 2, pp. 35-42.

Hutchins, B. (1976) "Project 1 – Non-Linear Transfer Function," *Electronotes*, Vol. 8, No. 68, pp. 3-8.

Le Brun, M. (1979) "Digital Waveshaping Synthesis," *Journal of the Audio Engineering Society*, Vol. 27, No. 4, pp. 250-266.

Mathews, M. (1969) *The Technology of Computer Music*, MIT Press, Cambridge.

Moore, F. R. (1978a) "An Introduction to the Mathematics of Digital Signal Processing, Part I," *Computer Music Journal*, Vol. 2, No. 1, pp. 38-47.

Moore, F. R. (1978b) "An Introduction to the Mathematics of Digital Signal Processing, Part II," *Computer Music Journal*, Vol. 2, No. 2, pp. 38-60.

Risset, J. C. (1969) *Catalog of Computer-synthesized Sounds*, Bell Labs.

Schaefer, R. W. (1970) "Electronic Musical Tone Production by Nonlinear Waveshaping," *Journal of the Audio Engineering Society*, Vol. 18, pp. 413-417.

Suen, C. Y. (1970) "Derivation of Harmonic Equations in Nonlinear Circuits," *Journal of the Audio Engineering Society*, Vol. 18, No. 6, pp. 675-676.