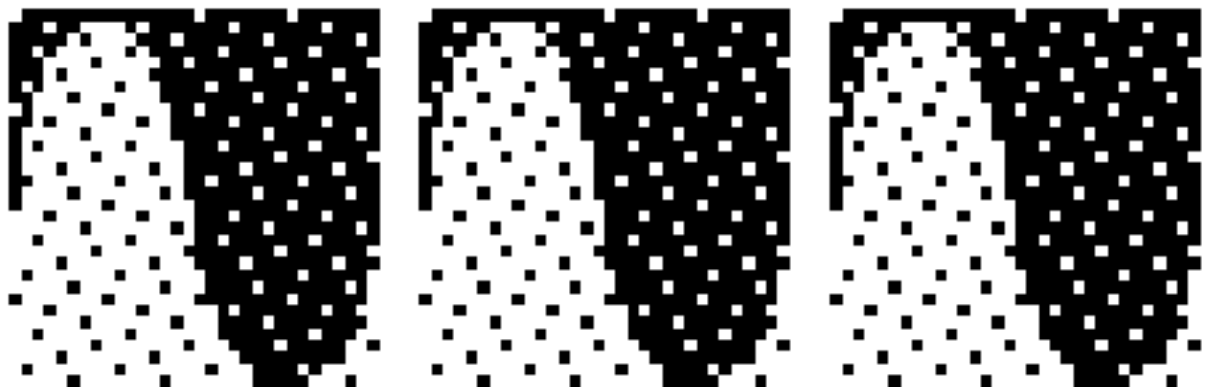


SoundHack

user's manual

tom erbe

center for contemporary music
mills college



SoundHack 0.66

©1992 Tom Erbe
Center for Contemporary Music
Mills College
Oakland, CA 94613
tom@mills.edu

May 4, 2025

I am writing this program to perform various sound file manipulations on the Macintosh that have previously not been available. At the present, I have implemented sound file type conversion, a varispeed/sample rate converter, sound file convolution, ring modulation, the phase vocoder, a binaural filter and an amplitude analysis and gain change module. In the near future I hope to add a spectral dynamics processor, and Larry Polansky's mutation synthesis. SoundHack can now read and write the following formats: Sound Designer II, Audio IFF, IRCAM, DSP Designer, NeXT .snd (or Sun .au) and TEXT. It can read (but not write) raw data files. It can read and write 8-bit μ Law, 8-bit linear, 32-bit floating point and 16-bit linear data encoding.

This program only works on Mac II and above machines! That is: you need a 68020 + 68881, a 68030 + 68882 or a 68040.

Thanks to Dr. Durand Begault of NASA-Ames for letting me use his binaural filter coefficients, Dan Ellis of MIT Media Lab for helping me with the Csound analysis feature and also to the Center for Contemporary Music for sponsoring my activities.

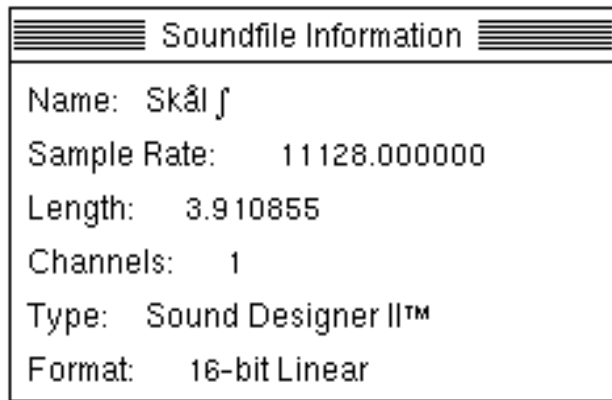
Please write me if you have any problems or suggestions!

Tom Erbe

• FILE MENU

Open...

opens sound files with the following file types: Sound Designer II or Audiomedia (Sd2f), Audio Interchange File Format (AIFF), those created by SoundHack in the IRCAM file format (IRCM) and DSP Designer (DSPs). Once the file is open the Sound file Information dialog box appears. This dialog box gives the name, sample rate, length in seconds, number of channels, type and numeric format of the sound file. This file is used as the input sound file to the processes under the Hack menu. Only one input sound file may be open at a time.



Open Any...

opens any file as a sound file. Useful for opening sound files that were created on other machines and have no file type. The files that it recognizes are: NeXT sound files, Sun sound files, IRCAM sound files (if not created by SoundHack) and text files. All other files are opened as headerless sound files. Text sound files should be formatted so that each line is a fixed point sample. Headerless files (both text and raw) must be saved to another format before being processed.

Close

Obvious.

Save A Copy...

Saves a copy of the opened sound file in any sound file format.

Quit

Obvious.

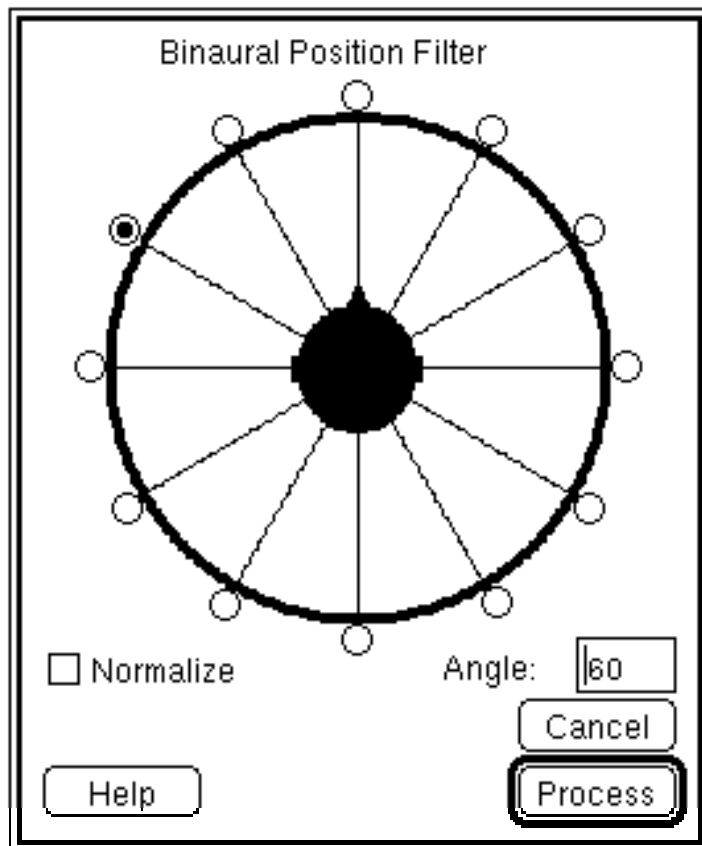
• EDIT MENU

The usual stuff, not too useful in this program though.

• HACK MENU

This is where all the sound file processing is. Most of the things in this menu involve a lot of calculations, and take time! A RISC based Macintosh is recommended (unfortunately they don't exist). I am writing this program not for fast routines, but for interesting ones. SoundHack pretty much takes over your Mac to do these, so if you think your Mac has frozen up, it probably hasn't. The Phase Vocoder is the worst of these processes in terms of taking over your Mac. So have patience.

Binaural Filter...



This process allows you process a monaural sound file (creating a stereo file) so that the result puts the mono signal at a simulated position around ones head. This is done by using a HRTF (head related transfer function) as a filter, one for each position around the head. To use, either enter the desired position in the "Angle" dialog box (in degrees) or click the appropriate radio button. This processing module has filter data for 12 positions around the head, but if you enter an angle between two positions you will get a filter which is the mix of the two filters around it. Check the "Normalize" box if you want the output to be normalized (brought to the loudest possible level) after computation. These binaural filters are sample rate conversions

(for 44100 samples per second) of those used in the dissertation "Control of Auditory Distance" by Durand Begault, which are approximations of the averaged monaural transfer functions shown in Blauert's "Spatial Hearing".

Convolution...

Convolve with Impulse Response
Impulse File: sine 1kHz
Sample Rate: 44100.000000
Channels: 1
Length Used:
**kiloBytes Needed
to Process File:** 198
Filter Gain: ☐ Low ☐ Med ☒ High
Impulse Window:
☐ Ring Modulate ☐ Moving
☐ Normalize

This process takes two sound files, the input file and the impulse response file and multiplies the spectrum of the two files together, producing a new sound file. The effect is a type of cross-synthesis, in which frequencies which the files have in common are reinforced. The "Length Used" window allows you to designate how much of the impulse response file to use. The "kilobytes Needed To Process File" number is an estimate of the size SoundHack needs to be set to. If you want to use large impulse responses, that is, one that causes this number to go over 900, you will have to quit SoundHack and reset the application memory size. The "Filter Gain" button should usually be set to "High", but if the impulse and input have similar spectrum, it should be set to "Med" or "Low" to avoid a clipped output. If a clipped output still seems unavoidable, save the output in NeXT floating point format, then use the Gain... module to normalize it back to an integer format. The "Impulse Window" choice will apply a selected envelope onto the impulse before convolution. This will give us smoother convolution, especially with a moving impulse window. A rectangular window is the same as no window at all. Check the "Normalize" box if you want the output to be normalized after computation. The "Moving" box moves a window through the impulse file, taking a new impulse after every block of processing. This window is the size set in the "Length Used" field. It is set so that the impulse file reaches the end at the same time that the input file reaches the end. For example, your input file is 10 seconds long, your impulse file is 5 seconds long and you have set "Length Used" to 1.0 for one second impulse response windows...

Input File

A	B	C	D	E	F	G	H	I	J
---	---	---	---	---	---	---	---	---	---

Impulse File

a	c	e	g	i					
b	d	f	h	j					

Output File

A*a	B*b	C*c	D*d	E*e	F*f	G*g	H*h	I*i	J*j
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

(Output frames actually overlap, but I couldn't think of how to show this clearly)

The first 1.0 second frame of the input file (A) will be convolved with the first 1.0 second frame of the impulse response file (a). Then the window on the input file is moved 1.0 second forward to B, but the window on the impulse response file is moved only 0.5 seconds to b. This is so both files will reach their ends at the same time. (Actually, the impulse response file reaches the end first and the last impulse response is zero-padded). In the other case, when the impulse response file is longer than the sound file, sections of the impulse response file will be skipped over. It is a good idea to set gain to "Low" if using the moving impulse (or save things in a floating point format), as the scaling is fairly unpredictable. The "Ring Modulate" box does a ring modulation (or convolution in frequency) between the two sound files.

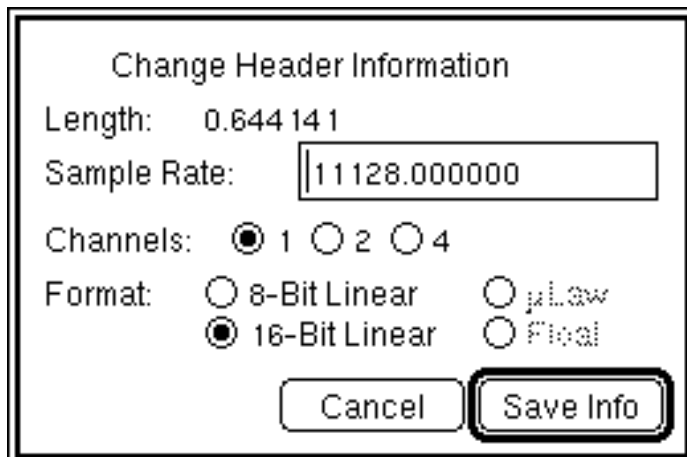
Gain Change...

Amplitude & Gain Change		
<input checked="" type="radio"/> dB <input type="radio"/> Fixed	Channel 1	Channel 2
Peak Value:	-4.976574	-INF
Peak Position:	5120	0
RMS Value:	-29.534100	-INF
Gain Factor:	<input type="text" value="4.976574"/>	<input type="text" value="INF"/>
<input type="button" value="Analyze"/>	<input type="button" value="Cancel"/>	<input type="button" value="Change Gain"/>

The equivalent functions in Sound Designer II and Alchemy are faster, so in most cases you will want to use those. However, my module will give you an RMS value,

and will allow a different gain factor for each channel, plus will work on floating point and μ Law files. Click on "Analyze" and the peak, peak position (in samples) and RMS values will be calculated. The gain factors will be set to normalize both channels independently. "Change Gain" will create a new file adjusted by the gain factors set. If you are dealing with a monaural file, only the channel 1 information is applicable.

Header Change...

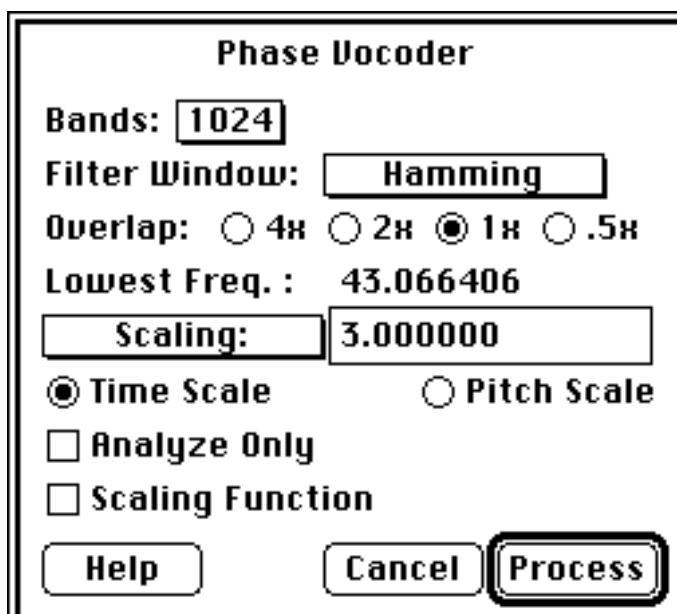


A dialog box titled "Change Header Information". It contains the following fields and controls:

- Length: 0.644 14 1
- Sample Rate: 11128.000000
- Channels: ☒ 1 ☐ 2 ☐ 4
- Format: ☐ 8-Bit Linear ☐ μ Law
☒ 16-Bit Linear ☐ Float
- Buttons: Cancel, Save Info

Allows you to change the sample rate, number or channels, and data format of the file open. If you open a headerless file, you should use this dialog to set things properly before saving a copy!

Phase Vocoder



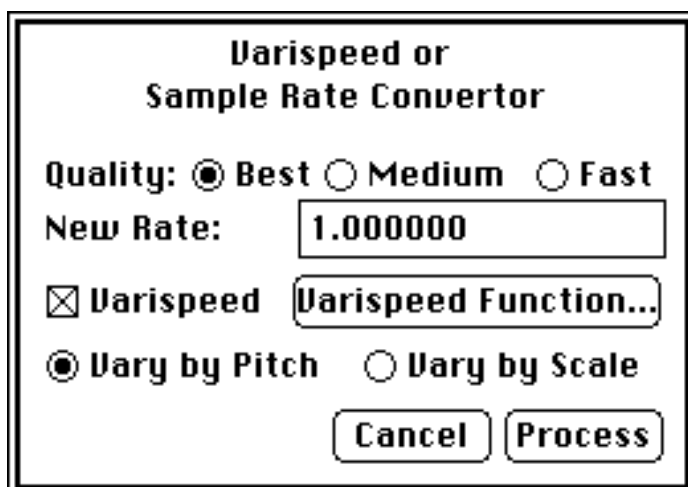
A dialog box titled "Phase Vocoder". It contains the following fields and controls:

- Bands: 1024
- Filter Window: Hamming
- Overlap: ☐ 4x ☐ 2x ☒ 1x ☐ .5x
- Lowest Freq. : 43.066406
- Scaling: 3.000000
- ☒ Time Scale ☐ Pitch Scale
- ☐ Analyze Only
- ☐ Scaling Function
- Buttons: Help, Cancel, Process

This process allows you to change pitch without changing the length of the sound file or to change length without changing pitch. It does this by extracting amplitude and phase information for 16-8192 frequency bands with a bank of filters. If time stretching is desired these phase and amplitude envelopes are lengthened (or shortened, for time compression), and then given to a bank of oscillators with corresponding frequencies to the filters. For pitch shifting, the envelopes are untouched, and given to a bank of oscillators with frequencies related by the pitch ratio.

To use the phase vocoder, set the number of "Bands" to the number of filter-oscillator pairs you would like to use. If you set this to be larger, you will get better frequency resolution, if you set it smaller, you will get better time resolution. The "Filter Window" menu allows you to choose different pre-FFT windows for different filtering characteristics. Only the Hamming, Hanning and Kaiser will give good results (the others are there only because I wanted to use a single menu throughout the program for all window selection). The "Overlap" setting adjusts the size of the filter window (relative to the number of filter bands) for analysis and synthesis and thus, the sharpness of the filter. A large setting (4x) will give the sharpest filter. The sharper filter will differentiate better between frequencies which are between bands, but responds to amplitude changes slower. Click the "Time Scale" button if you want time scaling, "Pitch Scale" for pitch scaling. Type the scale factor in the "Scale" box. Click on the word "Scale" (a pop-up menu) if you want to specify time scaling by the length desired, or pitch scaling by equal tempered semitones. Click the "Analyze Only" box to produce Csound compatible pvoc analysis files. If you want the time expansion factor or the pitch transposition factor to change during processing, click the "Scaling Function" box, and the "Draw Function..." button. This will bring up the Draw Function dialog, which is described later.

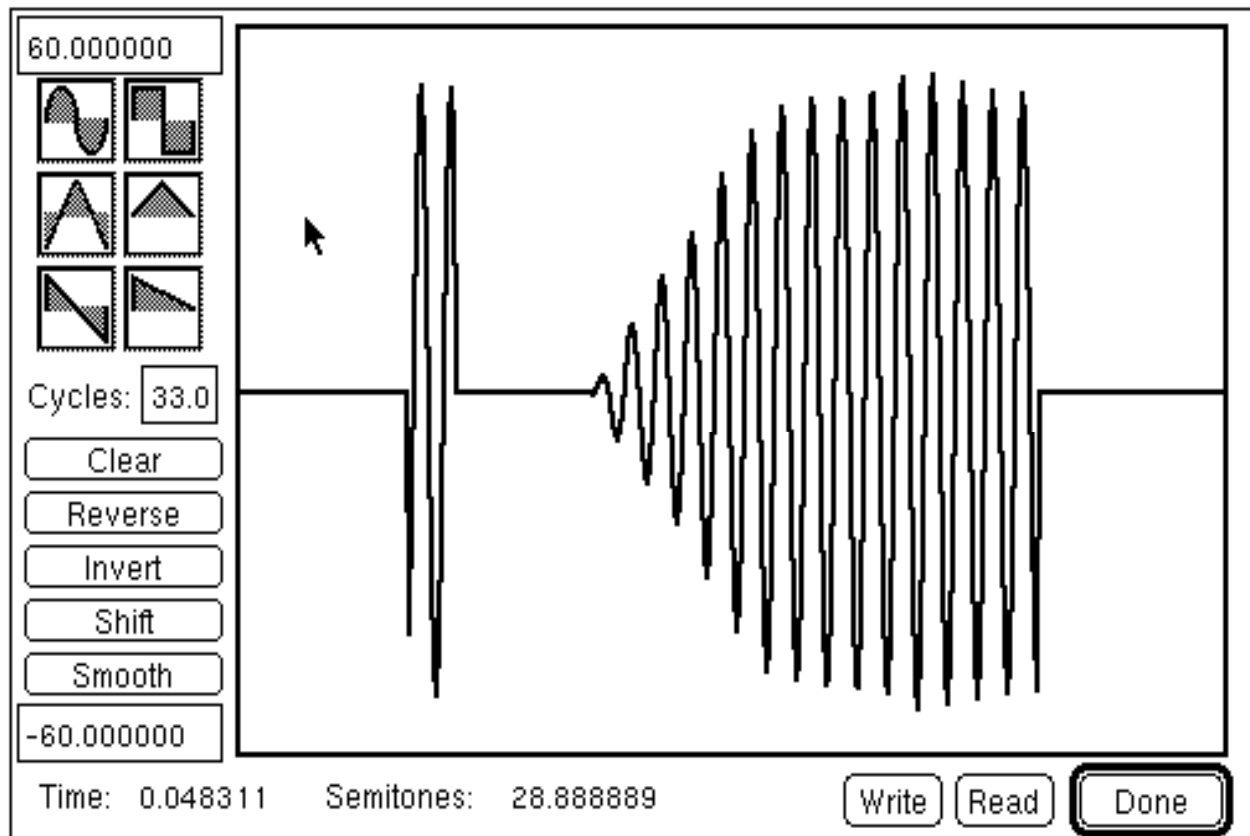
Varispeed...



As a sample rate converter, this is slower and maybe less accurate than the Sound Designer II orAlchemy software, so most of you won't use that function. Instead, click

the "Varispeed" box and a "Draw Function..." button will appear. This will bring up the Draw Function... dialog, giving you control over a 10 octave varispeed. The "Quality" buttons give you control over the size of smoothing filter used, and the resultant quality of interpolation/decimation. The "Vary by Scale" and "Vary by Pitch" buttons allow you to draw a curve for either pitch or scaling.

Draw Function...



This is my dialog for creating control functions. You can draw in the function window, set upper and lower limits for the function, clear, invert, reverse, smooth or shift the function, or use some of the customary wave forms (up to 100 cycles) for your control function. If the function window is cleared (with a constant value through the center of the screen), the wave form icon buttons will draw the wave form as the new control function. However, if the function window is not cleared, the wave form icon buttons will modulate (four quadrant AM) the existing control function. There is no facility for selecting, copy, paste or cut. You can read or write control functions as sound files. The "Time:" refers to the time in the input sound file and the other legend (currently "Semitones:") is updated depending on what the control function is controlling. There are only 400 points in the control function (though interpolated on) and that is the main limitation of this dialog box.

• CONTROL MENU

Show Output

This will show your sound whenever SoundHack writes sound to your output sound file (except during file copying and normalization).

Pause Process

This allows you to pause any long processing job in case you need to use your Mac, but you don't want to start all over with your processing. If you are running a convolution, it sometimes takes a while to pause (up to 3 minutes...).

Continue Process

This will pick up your processing where you left off.

Stop Process

This will kill your process and close the output sound file.

•BUG FIXES AND REVISIONS

.66 Added windowing selection for both convolution and the phase vocoder. Added "Vary by Pitch" / "Vary by Scale" selection in varispeed.

.65 Fixed crashing at the end of long varispeed calculations, cleaned up memory allocation (especially for convolution), added ring modulation (spectral convolution), cleaned up dialog annoyances. Made output window resizable, and made it white on black (like a scope). Changed software to shareware, too few people sent me music when software was musicware. Added bibliography screen.

.64 Added varispeed processing, added many things to the function dialog.

.63 Added TEXT sound file format. Added interpolation, sinc and number of cycles to function dialog. Added ramp enveloping of impulse to convolution. Added popup menus.

.62 Added function window for multirate phase vocoder.

.61 Sped up pitch transposition significantly, allowed pitch transposition by semitone and time warping by desired length.

.60 Added the phase vocoder, phase vocoder Csound analysis and show output. Fixed problem with normalization after moving convolution.

.59 Added moving convolution.

.58 Fixed problems with 8-bit AIFF files.

.57 AIFF files used to read everything from SSND chunk to EOF, now only SSND chunk is read. The Binaural processor often destroyed AIFF headers, I think I have this fixed. DSP Designer files are now written as well as read (though I have no way of checking, please DSP

Designer users, give me feedback).

- .56 Filter files are now closed properly after convolution.
- .55 Normalization feature disabled Binaural filtering, now fixed. Filter sensitivity added to Convolve.
- .54 Normalize after processing feature added. Dialogs adjusted for small monitors.

Previous to .54 I have little memory. 0.7 will have Larry Polansky's mutation functions added. 0.8 will have a spectral dynamics processor (useful for removing broad band noise, hum, etc.). FFTs will be sped up by the 56001 (if you have a Sound Accelerator or Audiomedia) eventually, and someday a small scripting facility will appear. 1.0 will be the last version. Csound support will disappear as soon as I write CsoundAnalysis (a Macintosh envelope for pvanal, lpcanal and hetro).

This software is shareware, if you use it, send me \$30.00 or some music that you have created. My address is: Tom Erbe, 360 Arlington Street, San Francisco, CA 94131.

•BIBLIOGRAPHY

- Apple Computer, Inc., *Inside Macintosh, Volume 1-6*, Addison Wesley, Reading, Mass., 1985-91.
- Begault, Durand, *Control of Auditory Distance*, dissertation, University of California, San Diego, 1987.
- Blauert, J., *Spatial Hearing*, MIT Press, Cambridge, Mass., 1983.
- Dolson, Mark, "The Phase Vocoder: A Tutorial", *Computer Music Journal* 10:4, 1986.
- Ellis, Dan, "pvanal.c", part of the Csound distribution, MIT, 1991.
- Gordon, J. W. and Strawn, J., "An Introduction to the Phase Vocoder", *Digital Audio Signal Processing: An Anthology*, editor J. Strawn, Kaufmann, Los Altos, Calif., 1985.
- Mark, David and Reed, Cartwright, *Macintosh C Programming PRIMER, Volume I*, Addison Wesley, Reading, Mass., 1989.
- Moore, F. Richard *Elements of Computer Music*, Prentice Hall, Englewood Cliffs, NJ, 1990.
- Polansky, L. and McKinney, M. "Morphological Mutation Functions: Applications to Motivic Transformations and to a New Class of Time-Domain Cross-Synthesis Techniques." *Proceedings of the ICMC*. Montreal, 1991.
- Reed, C. E. and Passin, T. B., *Signal Processing in C*, John Wiley, New York, NY, 1992.
- Vaseghi, S. and Frayling-Cork, R., "Restoration of Old Gramophone Recordings", *Journ. AES*, 40:10, 1992.