THE OTHER EAR : A MUSICAL SONIFICATION OF EEG DATA

John A. Dribus

Center for Experimental Music and Intermedia
University of North Texas College of Music
P.O. Box 331367, Denton, TX, USA
jad0054@unt.edu

ABSTRACT

The human auditory system is capable of assimilating large amounts of information in short periods of time. However, in order to allow that information to be understood, it must be delivered in such a way that much of it can be readily grasped. Through careful mapping of data streams to common musical parameters often found in electronic composition, *The Other Ear* presents EEG data in a form that should be accessible to listeners who are at least somewhat familiar with this compositional idiom.

1. INTRODUCTION

The Other Ear was realized through the use of an additive synthesis engine, by mapping the provided EEG data recorded from a subject listening to music. The goal of this project was to present the data accurately in regard to its temporality and other features, while simultaneously rendering it as an aesthetically pleasing musical composition.

The usual compositional elements such as pitch, texture, form, and timbre were given serious consideration. Since each of these musical elements (and others) had to be generated and driven by the data provided, careful consideration of the shape and contour of each data file was considered prior to mapping. Data channels were mapped to sonic parameters based upon the appropriateness of the data plot's contour to the unique function of musical element being generated. For example, in this sonification it seemed most effective to map irregular, rapidly changing data files to panning functions, while more linear and static data files were mapped to frequency and volume. The distribution of data channels among the different musical parameters was also determined in part by the design of the synthesis engine, which will be described in detail here.

2. TECHNICAL CONSIDERATIONS

The mapping process was executed by an additive synthesis program originally written for musical creation. Numerous modifications were necessary to render it suitable for this project. The details of the mapping process are found below.

2.1. Synthesis Engine Architecture

The additive synthesis engine was written in MAX/MSP and was capable of real-time generation of audio files. This capability was extremely important during the sonification process, as parameters could be modified and tweaked in real time. Due to the need for real-time processing, the number of oscillators used in the synthesis was limited to twenty by the processing capabilities of the MacIntosh 1.25 Ghz dual-processor G4 that was used for writing the audio files. The basic construction of the synthesis program is shown in Figure 1. The twenty oscillators in the bank were grouped into summed pairs before being sent to the delay and panning patches. The output of the synthesis engine was recorded as stereo sound files.

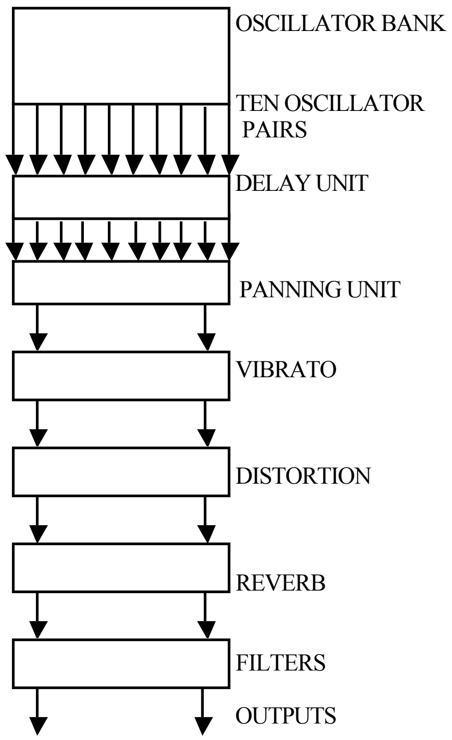

Figure 1. *Basic components of the additive synthesis engine.*

2.2. The Data Conversion Process

Before the data could be used in MAX/MSP, each EEG data file was read into a table and converted to an audio file using Csound. Further processing included DC offset removal and normalization. The resulting files were saved as 44.1Khz audio files and were each 3.401361 seconds in length. The change in file-length from 5 minutes to ~3.4 sec. was due to the conversion from 500Hz to 44.1Khz sampling rate. After each data/audio file was read into a buffer in MAX/MSP, a read rate of .011338 was used in order to restore the data's temporal integrity, yielding a read time of exactly 5 minutes.

2.3. Frequency Mapping

A majority of the EEG data files were used to determine the frequency of the oscillators. For most stereo pairs of audio files recorded, two EEG data files were used. The data files were placed in their channels based upon the approximate locations of their source in the brain in relation to the location of the speakers in the Sydney Opera House Studio. Table 1 details the mapping of audio parameters to different stereo pairs. The intended effect was to create the impression of hearing from inside the head.

AUDIO CHANNEL OR STEREO PAIR	FILTER DATA	PAN DATA/ DIRECTION	FREQUENCY DATA
Speakers 1&6	28 (VNVB)	33 to 28, front/back	Fz, Oz
Speakers 2&5	30 (HNHR)	33, front/back	F3, P3
Speakers 2&10	29 (HPHL)	28 to 29, left/right	Fp1, Fp2
Speakers 3&9	27 (VPVA)	29, left/right	F7, F8
Speakers 4&8	33 (Mass.)	28 to 29, left/right	T5, T6
Speakers 5&7	32 (OrbOcc)	29, left/right	O1, O2
Speakers 7&10	29 (HPHL)	33, front/back	P4, F4
Speakers 11&13	32 (OrbOcc)	33, front/back	Fcz, Pz
Speakers 12&14	29 (HPHL)	29, left/right	Fc3, Fc4
Speaker 15 Zenith	32 (OrbOcc)	None	Cz, modulated by Erbs
Speaker 16 Sub	29 (HPHL)	None	ECG

Table 1: *Data mapping by audio channel or stereo pair. Speaker numbering adheres to the diagrams of the Sydney Opera House Studio that were provided on the ICAD website.*

If a stereo pair of speakers was located in a left/right orientation, frequency data from both the left and the right source data files was mixed in both speakers. Although this placement potentially blurs the discrete locations of the data sources, it also allows for dynamic panning effects that would have otherwise been impossible. In most cases, the data mapped to both channels in a stereo pair is similar enough that the blurring of location is not perceptible, but the panning certainly is noticeable.

For each oscillator, the frequency produced at any given moment was determined by:

$$f = (x+o)smp \qquad (1)$$

where x is the number received from the normalized data file, o is an offset value to eliminate negative values, s is a scaling value, m is a frequency multiplier, and p is the partial number. The frequency calculations were performed at the audio rate. The scaling value was derived by taking the ratio of the first value found in the chosen data file (for Fp1 this was –88347) to the first value found in the data file beginning with the largest value (this was P4, at –62506). In the case of Fp1, for example, the ratio was 0.71. A chart of scaling values can be found in Table 2. The frequency multiplier of 15 was found through trial and error in an attempt to assure that all speakers had signal in an audible range throughout the course of the sonification. Finally the frequency value was multiplied by the partial number to ensure a harmonic placement of partials. Since each stereo pair of oscillators received data from both of the frequency files in use, and any given oscillator could only process data from one source file at a time, each output file contains the first ten partials of two harmonic sounds.

DATA FILE	ASSIGNED PARAMETER	CHANNEL # FOR AUDIO PLAYBACK	INITIAL VALUE	SCALING VALUE
ch01-Fp1	frequency	2	-88347	0.71
ch02-Fp2	frequency	10	-86552	0.72
ch03-F7	frequency	3	-244209	0.26
ch04-F3	frequency	2	-227768	0.27
ch05-Fz	frequency	1	-292070	0.21
ch06-F4	frequency	10	-234542	0.27
ch07-F8	frequency	9	-250256	0.25
ch08-FC3	frequency	12	-84026	0.74
ch09-FCz	frequency	11	-164212	0.38
ch10-FC4	frequency	14	-278940	0.22
ch11-T3	Not used			
ch12-C3	Not used			
ch13-Cz	frequency	15	-90263	0.69
ch14-C4	Not used			
ch15-T4	Not used			
ch16-CP3	Not used			
ch17-CPz	Not used			
ch18-CP4	Not used			
ch19-T5	frequency	4	-138264	0.45
ch20-P3	frequency	5	-122820	0.51
ch21-Pz	frequency	13	-64333	0.97
ch22-P4	frequency	7	-62506	1
ch23-T6	frequency	8	-107651	0.58
ch24-O1	frequency	5	-178120	0.35
ch25-Oz	frequency	6	-63952	0.98
ch26-O2	frequency	7	-71410	0.88
ch27-VPVA	filter sweep	3,9	Not used	1
ch28-VNVB	filter sweep	1,6	Not used	1
28, cont	pan	1,2,4,6,8,10	Not used	1
ch29-HPHL	filter sweep	2,7,10,12,14,16	Not used	1
29, cont	pan	2-5,7-10,12,14	Not used	1
ch30-HNHR	filter sweep	2,5	Not used	1
ch31-Erbs	freq. modulator	15	Not used	1
ch32-OrbOcc	filter sweep	5,7,11,13,15	Not used	1
32, cont	hit duration	ALL	Not used	1
ch33-Mass	filter sweep	4,8	Not used	1
33, cont	pan	1,2,5-7,10,11,13	Not used	1
ch34-EDA	volume	ALL	Not used	2
ch35-Resp	trigger	ALL	Not used	10
ch36-ECG	frequency	16	Not used	0.25

Table 2: *Data file assignments by parameter.*

2.4. Frequency Exceptions

The sub channel was treated differently from the stereo pairs. It was given only one data file (ECG) to determine its frequency. The ECG file yielded a sound that was clearly recognizable as a heartbeat. A special scaling value of .25 was applied in order to match the sub's frequency response and to provide a consistent low frequency component to the sonification.

The zenith speaker was also treated as a mono channel. The Cz data file was used to generate its frequency values, and a special scaling value of 2 was used to ensure that the speaker would always produce relatively high frequencies. The frequency data from the Cz file was modulated by data from the Erbs. Since the Erbs data was very similar to the ECG data used for the sub channel, there was a direct relationship, though not stereo pairing, between the sub and the zenith speaker. Together, these two speakers create a rhythm otherwise lacking in the sonification.

2.5. Amplitude Determination

In order to achieve sounds with some musical characteristics, the amplitude value for each oscillator came from recorded samples of musical instruments. This involved doing an FFT analysis of a sound and recording the amplitudes of its strongest partials. Sounds used were from wind instruments, percussion instruments and the human voice. Ten amplitude values were sent to the oscillator bank, corresponding to the number of partials present for each frequency source file. (For consistency, the channels (sub and zenith) that were not using paired frequency data sources were still given amplitude values for ten partials, although twenty would have been possible.)

To compensate for the rather dull sounds that might be produced due to the lack of high partials, the oscillators read natural waveforms from the aforementioned sampled sounds instead of the more typical sine functions. In many cases, different waveform and amplitude value sources were used in the same oscillator, yielding complex hybrid timbres. The pairing of waveform and amplitude was done through trial and error, and only the best nine available combinations were used. The timbres were classified as gentle, medium or harsh, depending upon the amount of noise in the signal. Timbre was used to help articulate the form of the work during the sonification process, especially during the middle third of the work.

2.6. Mapping of other musical parameters

Many of the more irregular data files were used to drive musical parameters other than frequency. Table 2 details the allocations of all the data files. Overall volume levels for the work were determined by the inverted EDA data stream. This produced an upward dynamic curve which had rapidly changing dynamic levels near important formal junctures. Since the volume coefficient was applied at each oscillator, other incidental factors including reverb feedback and distortion tended to influence the actual volume of the output, generally exaggerating the dynamic contour provided by the EDA. The frequency response of the speakers was also taken into account here, as the lower frequencies at the outset of the sonification would be perceived (on most sound systems) as softer than their dB level might indicate. This feature also worked well into the intended formal affect of the work by causing the low frequencies at the outset to grow gradually in intensity as the frequency level rose.

The data files VNVB, HPHL, and Mass were used to drive the stereo panner built into each pair of oscillators. These files were chosen primarily because of their rapid changes in direction. Each file also had more variation towards the latter part of the sonification, which became an important formal factor. HPHL was designated as the left/right panner, and Mass. was used for front to back panning. VNVB was applied to a few channels later in the work (See Table 1) when a less homogenous panning structure was desired.

The three data files that were used for panning were also mapped to filtering functions, along with VPVA, HNHR, and OrcOcc. Filtering was only active in the last third of the piece, and was used to add timbral variety to the sustained sound mass found in that section. Filtering files were selected on basically the same criteria as were panning files, allowing for fast, gestural frequency sweeps.

The primary formal features of the sonification were triggered by the Resp. data file. This file was coupled with an amplitude tracker and was used as a trigger. The huge dip in this data plot at about 1:33 corresponded temporally to a similar dip in the inverted EDA file used for volume control. This point was used to articulate the first sectional division in the sonification, and later events were triggered by quasi-random number functions that were initiated at this point as well.

3. MUSICAL OVERVIEW AND FORMAL CONSIDERATIONS

The data mapping process yielded a sonification in three sections. The first and third sections consist of more-or-less continuous sound. The middle section is pointillistic in character, eventually building in texture and timbre into a third section. Careful musical decisions were made in order to articulate a clear musical form while maintaining the temporal integrity and other implications of the data. In mapping the data, special attention was given to exceptional events, especially if they occurred in several of the data plots. A prime example of this is the dip in the Resp data plot near 1:35. The following description highlights some of the sonification's musical characteristics.

3.1. Section 1 (0:00 to 1:33)

Most of the data plots used for frequency determination start near their lowest levels. The very low sounds of the sub and the higher sounds from the overhead speaker add much to the timbral character of this section. A gradual rise in pitch and volume characterizes this opening. There is noticeable frequency variation in the front stereo pair (Fp1 and Fp2), and the pitch here starts at a higher level than in most of the speakers, making the panning here more evident. Therefore, the focal point of the opening is the front part of the room. By the time that the Resp data file amplitude tracker initiates the second section, all the frequency levels have risen to a point where stereo panning is more audible and individual frequency characteristics of each stereo pair are easier to detect. About 30 seconds before the Resp channel triggers the second section, there is a small dip in its intensity (its only other prominent feature during the course of the work.) This is used to trigger a heavy reverb that gradually comes into full effect over about 25 seconds. The heavy reverb provides a smooth decay when the second section begins with silence in all but the sub channel's oscillator bank. Table 3 details musical events as they occur in the sonification.

TIME	EVENT	TRIGGER	TIMING
0:00	start	key	
1:05	heavy reverb (except sub)	Resp. peak 1	25 sec. fade-in
1:33	oscillator volume to 0 (except sub)	Resp. peak 2	5 sec. fade-out
“	light vibrato	“	10 sec. fade-in
“	low res filters to 4000hz	"	instant
"	change pan source in some channels	"	instant
1:35	defualt (light) reverb	2.5 sec. delay from Resp. peak 2	10 sec. fade-in
1:38	"gentle" timbre setting	5 sec. delay from Resp. peak 2	instant
"	random attack increment set, enveloping initiated	"	instant
1:43	sub channel fade-out	10 sec. delay from Resp. peak 2	10 sec. fade-out
Variable	"medium" timbres set	hit increment < 3 sec.	instant
"	all timbres set	hit increment < 1.5 sec.	instant
"	delay loop initiated	"	20 sec. fade-in
"	attack envelope steepened	"	30 sec. fade-in
"	delay feedback to 50%	"	30 sec. fade-in
"	"harsh" timbres set	hit increment < .75 sec.	instant
"	delay loop stopped	hit increment < .25 sec.	instant
"	attack envelope back to default	"	instant
"	heavy vibrato	"	7 sec. fade-in
"	moving resonant filters initiated	"	instant
"	"gentle" timbre setting	"	instant
"	medium reverb set	30 sec. delay from hit increment to < .25	7 sec. fade-in
"	medium-heavy reverb set	40 sec. delay from hit increment to < .25	7 sec. fade-in
"	heavy reverb set	50 sec. delay from hit increment to < .25	7 sec. fade-in
4:45	all oscillators fade-out	clock	5 sec. fade-out
4:51	final hit	clock	9 sec. fade-out
"	resonant filters to 16000 Hz	"	1 sec. fade-in

Table 3: *Musical events as they occur throughout the sonification.*

3.2. Section 2 (1:33 to ~ 4:00)

The opening of the middle section is marked by the heart-like sound from the ECG data heard through the sub channel. This sound fades after a few seconds and then the sub channel is treated like the others. A windowing function is initiated in this section, and it causes small bursts of sound to be heard at different times in different stereo channels. The timing increments between each successive sound burst are determined by a random time value generated for each stereo channel ranging from three and five seconds. Each time a sound burst occurs, the separation increment between successive windowing functions in the given channel is decreased by 3% (5% in the sub, because it starts this section about 15 sec. late). The windowing function can be thought of as a simple turning up or down of the volume level of certain oscillators at various times. Although sound is not heard in all channels simultaneously, the processes that determine pitch are working constantly in all channels, causing any pitched sound that is heard at a given time to correspond correctly to the timing of the data file being used to generate frequency.

The triggering of various musical events in this section is determined on a per channel basis based upon the current attack separation interval in a channel (See Table 3). When the event separation increment for a channel reaches 1.5 sec., a delay loop is initiated. This produces an echo-induced thickening of texture by slightly delaying each attack by a different time value in 9 of the 10 oscillator pairs in each channel. These delayed events are diffused by the panning engine. Note the temporal integrity of the data files is also maintained here; each event begins when it should, and the delay loop simply prolongs the perception of the ever shortening bursts of sound. It should also be emphasized that the windowing functions applied in this second section are used to vary the character of the sounds heard over the course of the sonification and prevent a loss of listener attention due to otherwise inevitable monotony. Although the exact timing of each sound burst is determined through stochastic processes, the initiation of these processes is triggered by the data, and the sounds that are heard at any given time do derive their characteristics from the neural activity captured at the corresponding instant during the EEG recording process.

3.3. Section 3 (~4:00-5:00)

Each stereo pair arrives at Section 3 at a slightly different time, based upon the random time increment generated for Section 2. Section 3 is much like the first section, but the volume is higher, the pitch is higher, and sweeping filters add a dynamic dimension to the timbral characteristics of the section. At the end of the work the windowing function is reinitiated in order to draw the form to a close. At 4:51, a final attack is heard, and it fades out for the remainder of the sonification's duration.

4. CONCLUSION

The musical treatment of the EEG data files allowed this sonification to communicate the information contained in those files in a manner that was both audibly perceptible and aesthetically interesting. Through careful mapping decisions and synthesis engine architecture, the natural form of the data recorded from one musical listener was translated to a musical listening experience for other ears. Included in this zip archive are sixteen channels of audio, named by my surname plus their proper speaker number, according to the numbering convention. Soundfile “dribus16.wav” is a separate signal for the sub channel. Low frequency signal from other channels must not be mixed into the sub channel. The following sixteen soundfiles are included: dribus01.wav, dribus02.wav, dribus03.wav, dribus04.wav, dribus05.wav, dribus06.wav, dribus07.wav, dribus08.wav, dribus09.wav, dribus10.wav, dribus11.wav, dribus12.wav, dribus13.wav, dribus14.wav, dribus15.wav, dribus16.wav.