Sound Authoring Tools for Future Multimedia Systems

Marco Bezzi
CSC - Univ. di Padova

Giovanni De Poli
CSC - DEI Univ. di Padova
depoli@dei.unipd.it

Davide Rocchesso
DST - Univ. di Verona
rocchesso@sci.univr.it

Abstract

A framework for authoring non-speech sound objects in the context of multimedia systems is proposed. The goal is to design specific sounds and their dynamic behavior in such a way that they convey dynamic and multidimensional information. Sounds are designed using a three-layer abstraction model: physically-based description of sound identity, signal-based description of sound quality, perception- and geometry-based description of sound projection in space. The model is validated with the aid of an experimental tool where manipulation of sound objects can be performed in three ways: handling a set of parameter control sliders, editing the evolution in time of compound parameter settings, via client applications sending their requests to the sounding engine.

1. Introduction

The sonification of multimedia systems and the design of auditory icons are emerging research areas in the multimedia community [18]. Much of the interest in the topic stems from success obtained in human-machine interaction where visual objects, and speech/text communication are augmented with auditory information, which often come from spatial audio displays or from non-speech sounds [3, 20, 14]. In particular, musical timbres are well suited to fast communication of multivariate information, since they are inherently multidimensional entities and the hearing system does a terrific job in integrating and discriminating all the subtle variations of multiple dimensions. However, designing information-driven, dynamic sounds is a relatively new area of interest in the multimedia community. A pioneering work was done by William Gaver, the designer of the Macintosh SonicFinder and proponent of *auditory icons* based on sampled and synthesized sounds [11]. A rigorous methodology for sound design, based on the perception of musical timbres, has been recently introduced by Stephen Barrass [1]. He proposed an abstract Information-Perception Space inspired by the widely-used Hue-Saturation-Lightness Color Space, then deriving a specialization for auditory display, called the Timbre, Brightness, and Pitch Information-Sound Space (TBP ISS) [2].

Much of the knowledge of sound timbres and sound design comes from the studies of musicians, psychoacousticians and, as far as computer applications are concerned, computer music researchers. In the fifties, Pierre Schaeffer proposed a phenomenological descriptions of *Sound Objects* [24], categorized as entities differing in one or more perceptual properties. Schaeffer's classification was extended by Schafer [25], who also introduced a catalog of sounds organized according to referential attributes, i.e. properties of sound sources. As a result of these and other studies, we now have a consolidated lexicon for describing sound objects both from a phenomenological or a referential viewpoint. However, while decent definitions and models are available for some of the sound attributes, such as loudness, pitch and brightness [26], much effort has to be spent before reaching a satisfactory model of timbre as a whole. From a multimedia perspective, such a model would allow timbre categorization and synthesis of whatever sounds, based on navigation in a compact space. The best knowledge available in timbre research at this time is well summarized in [13], where it is stated that a comprehensive approach to timbre should embrace and expand two modes of perception: (i) The source mode, where we get an impression of an excitation *method* (e.g. plucking, bowing) interacting with some high-level *property* of a resonator (e.g. string, membrane); (ii) The interpretative mode, where tones can be related along perceptual dimensions, such as brightness, harmonicity, nasality, etc. . We like to think about this dichotomy as a distinction between sound *identity* (which is determined by a certain combination of source method and property) and sound *quality* (which is determined by a particular set of parameters describing the sound source and the transformation/reproduction paths). According to this representation of sounds, in sec. 2 we propose a general system architecture for sound design, which is based on three layers, two of which are aimed at working on sound identity and quality. The third layer is introduced to deal with the task of sound projection in space. As an experimental software system, in sec. 3 we describe a Sound Authoring Tool (SAT), a sort of container based on the three-layer architecture where we are gradually introducing as much of our

0-7695-0253-9/99 $10.00 © 1999 IEEE

sound-design knowledge as possible.

2. System Architecture

So far, the most rigorous approach to sound design for multimedia applications has been that of Barrass [2]. His TBP ISS is a versatile and compact model which can be used to sonify information of many kinds. The radial and longitudinal dimensions of the TBP ISS are bound to brightness and pitch, respectively, thus contributing to the description of sound quality. The timbral dimension of the model, giving the sound identity, is restricted to be a small collection of sound samples uniformly distributed along a circle derived from experiments in timbre categorization by Grey [12]. A pitfall of this model is that it is not clear how to enrich the palette of timbres by insertion of new samples. In fact, the Grey timbre space shows clustering between disparate physical systems such as a jet-driven flute and a bowed cello string. Since a time-frequency interpretation of the Grey space do exist, one should locate the new timbres in terms of properties such as high-frequency energy at onset and synchronicity in transients of upper harmonics. However, more recent investigations found that perceived sounds tend to cluster based on shared physical properties [13, page 266]. Moreover, since the sound-producing fundamental mechanisms are far less than all possible sounding objects, we find a timbre categorization based on physical descriptions more convenient. In the architecture that we are proposing, an important role is played by a set of physics-based blocks which can be connected in prescribed ways to give rise to a large variety of sound generators.

The First Layer of the system is responsible for establishing the identity of a sound object. Physical attributes are made available as handles for activating the sound source and controlling its timbral dynamics within a given category of timbre identity.

A Second Layer of the system is designed to accomodate all the signal-processing devices (typically, digital filters) specifically designed to modify the sound quality. To ease the task of the designer, these tools can be parameterized in terms of perceptual scales.

A Third Layer of the system is responsible for all the spatial sound attributes. The nature of spatial sound perception [5], and the specificity of algorithms used for altering it, are strong motivations for distinguishing this kind of processing from that of the second layer. In particular, the third layer accomodates algorithms for reverberation, spatialization, changes in sound-source position and size.

Summarizing, we are proposing a system organized into three communicating layers which correspond to three coarse attributes of sound objects:

- Timbre identity (Generation)
- Timbre quality (Modification)
- Spatial organization (Ambience)

As we have described, there is a perceptual justification to this organization which comes from the vast literature of psychoacoustics, especially timbre research and auditory scene analysis. There is also an engineering justification coming from forty years of computer music research, namely from the fact that various sound features are differently exploited by different sound synthesis and processing techniques. Therefore, we propose to use a different family of techniques for each of the layers, namely

- Physical modeling
- Spectral modeling
- Geometric modeling with perceptual interpretation

Incidentally, we also observe that the three-layer architecture is compliant with the existing working practice of sound engineers. During rehearsals, they use to arrange sound sources first, then the equalization stage is adjusted, and finally sound is spatially controlled by means of stereo panning and reverberation. The ergonomics of sound engineering has been validated by many decades of practice and is widely recognized to be very effective.

2.1. Generation

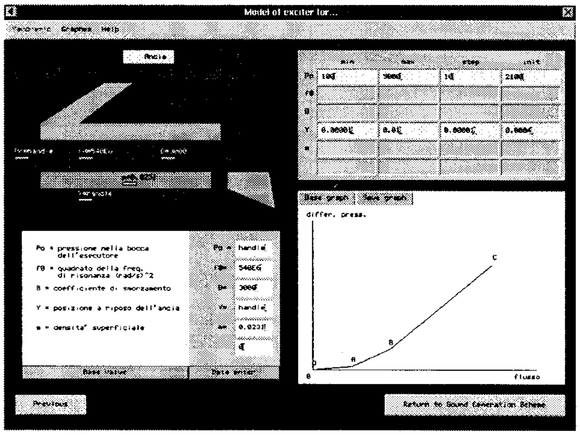
The Generation layer ir responsible for establishing the overall timbral identity of the sound object. The concept of sound identity is rather vague and depends on several timbre attributes. However, it is widely recognized that initial transients and articulation have great importance in the process of identification of timbral families [13]. These rapidly-changing sound attributes are produced by trajectories in the phase space of dynamic systems, and these trajectories are determined by the nature of the underlying physical phenomena. Different trajectories are likely to produce different percepts, so that we easily distinguish between bowed strings and jet-driven pipes, since the former is based on a nonlinear friction mechanism and the latter is based on a nonlinear fluid-dynamic mechanism. On the other hand, the discrimination of a violin from a viola is not that obvious and it is more a matter of sound quality of timbres sharing the same identity (and the same mechanism of sound production).

Along the line of these considerations, we chose physical modeling as a viable method for implementing the generation layer in our system. Nowadays, physical modeling is one of the most popular sound synthesis methods, due to the realistic dynamic behavior of the sound objects and to the intuitiveness of control of synthesis parameters. In fact, these parameters are directly related to physical attributes such as size, mass, stiffness, etc., which are the same that a performer have access to in a real instrument. The other side of the coin is the difficulty of implementing models which are general enough to cover broad families of timbres. Instead, dozens of models have been developed for simulating specific sound production mechanisms, such as reeds, air jets, bows, hammers, etc.. Therefore, we designed the generation layer as a container where we will be inserting our models as soon as they get developed or ported from other implementations. Within each model, parameters can be controlled in such an extent that substantial timbral modifications are possible without loosing the identity which is established by the modelled mechanism.

It might be thought that using different models for different sound families is a fundamental limitation to the freedom of the sound designer. However, experiences in computer music have shown that timbres are not easily conceived "in front of a white canvas". The designer largely benefits from reference models and, at the same time, these models help the listener with "landmarks" in the soundspace. Most of the sound generation mechanisms that can be conceived have inspired the design of musical instruments and useful mathematical descriptions of these mechanisms can be found in the musical acoustic literature.

Timbre selection in the Generation layer is categorical, since various model mechanisms can be chosen from a catalog. Timbre was a categorical dimension in TBP ISS as well, but in that case the choice was limited to a few (typically eight) timbres chosen from a catalog of samples. With our model-based approach to timbre design we can represent a much larger variety of timbres by means of modular composition of a relatively small number of basic blocks. Different model blocks can be composed into a single sound generation object by adopting a modular description of sound-producing systems. A common characterization of musical instruments outlines two main blocks: an exciter and a resonator [6, 10]. The exciter is usually a nonlinear system capable of initiating and sustaining the oscillation that takes place in the resonator, the latter being usually described as a linear system. In physical modeling, exciters are represented by means of lumped components (springs, dampers, masses, etc.), while resonators are represented by distributed descriptions (waveguides, partial differential equations, etc.). This decomposition is useful because different computational techniques can be used for resonators and exciters, and because different exciters can be coupled with different resonators, thus achieving modularity and enlarging the timbral space without expanding the number of models enormously. It is also useful to introduce a third block, whose purpose is that of interconnecting exciter and resonator by eliminating possible idiosyncrasies such as variable incompatibilities and instantaneous circular dependencies [6]. In the interconnection block we also insert those sound components which are difficult to model but are essential for ensuring naturalness to the sound. A typical example is that of pulsed noises, as they are found in many instruments. These noises are amplitude modulated by some physical variable (e.g. volume velocity in a reed instrument) and therefore they can not be attached to the sound as a postprocessing transformation. Sound residuals such as these have to be mapped into the physical model, just in the same way as textures are mapped onto 3D objects to enhance realism of a computer-modeled scene.

As an example of exciter for the sound generation layer we developed a dynamic reed model in the SAT application. A snapshot of the window is depicted in fig. 1. In the top-left region there is a schematic representation of the reed model. Below it, the parameters which are associated to physical attributes (mass, resonance frequency, damping, mouth pressure) can be either assigned with numerical values or set to `handle`, thus meaning that the parameters can be dynamically varied and mapped to higher-level sound attributes. In this latter case, the range of variability can be set in the top-right region of the window. In the bottom-right region there is a user-modifiable curve representing a nonlinear relationship between two physical variables, in this case the pressure drop and the air volume velocity across the reed. This nonlinear curve is present in most of the sound-production mechanisms and is responsible of much of the dynamic character of the sound object.



	min	max	step	init
Po	10	3000	10	2100
F0				
B				
Y	0.0000	0.01	0.0000	0.000
m				

Po = pressione nella bocca dell'esecutore — Po = handle

F0 = quadrato della freq. di risonanza (rad/s)^2 — F0= 540E6

B = coefficiente di smorzamento — B= 3000

Y = posizione a riposo dell'ancia — Y= handle

m = densita' superficiale — m= 0.0231

Figure 1: Reed model

2.2. Modification

The second layer of the proposed architecture provides means for affecting the quality of sound objects. These transformations are best referred to perceptual attributes of timbre in the steady state, such as brightness, harmonicity, etc. . Some of the transformations can be achieved by means of digital filters shaping the spectral content of sounds. Some other transformations, such as changing the degree of harmonicity, can be performed only after analysis of the sound coming from the generation layer.

In our schematic view, short-term features found in transients and articulation are ascribed to the generation layer, being peculiar of the underlying physical mechanisms. On the other hand, long-term features typically found in the steady state or during sound decay can be modified in a post-processing stage. Among the features having perceptual salience, only the brightness, or centroid of the spectral distribution, can be identified as a consistent dimensional attribute of timbre [13]. The brightness can easily be affected by changing the cut-off frequency of a lowpass filter, as it was done in [2]. There is less consensus on other spectral attributes affecting the quality of sounds. From experiments on timbre characterization based on tools borrowed from speech recognition [9], there is evidence of another attribute, someone calls it nasality [13], which is the relative amount of energy in the upper partials as compared to the lower. The nasality can be affected by changing the slope of the spectral envelope of a lowpass filter. Fig. 2 depicts how the two kinds of modifications can be achieved by means of a linear filter.

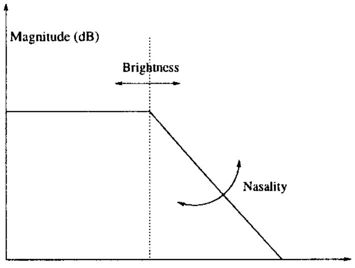


Figure 2: Spectral modifications of timbre quality by means of a lowpass filter

It is worth mentioning that changes in brightness and nasality can be obtained by direct control of the sound generation parameters. However, setting the parameters of a physical model is not an easy task and requires different skills from those expected from sound designers using a tool such as SAT. Generation models are best accessed by selection of instrument templates and fine parameter tuning based on trial-and-error. On the other hand, having direct access to meaningful perceptual parameters like brightness and nasality is a desirable feature which is easily provided in the second layer of our architecture.

2.3. Ambience

The third layer of the SAT architecture gives the way of specifying how the sound object should be located in a virtual 3D space and how the (virtual) space should affect the sound. Several degrees of accuracy can be used to specify the spatial attributes of sounds, from simple stereo panning up to detailed models of enclosures. In this context, we recommend using models which are simple enough to be computed in real time, and expressive enough to allow control on relevant parameters such as size, absorption, distance, direction, etc. . Models such as these have been studied [15, 22, 23] and extensively tested via interactive graphical environments [22]. It is important to present the physical/geometric parameters through a perceptual screen, in such a way that the sound designer have an intuitive idea about how the control actions will affect the sound object. Such a perceptual space has been defined after psychoacoustic testing [15], and has dimensions such as warmth, envelopment, liveness, etc. .

As far as location of virtual sources in space is concerned, the methodologies are highly dependent on the kind of display to be used (headphones, two or more loudspeakers), on the properties of the listening space and on the position of the listener [16, 19]. Therefore, using the same ergonomic considerations used by Barrass to assert that loudness is not a good sonification parameter [2, page 120], we assert that sound projection in space is not a robust design operation. For the purpose of sonification of multimedia applications, we suggest using simple spatialization based on models such as [21], where the loudspeakers are fed with delayed and attenuated copies of the incoming signal. Even though this model is not very accurate in spatial rendering, it proved to be quite robust to changes in positions of the listener. Moreover, Doppler effect due to movement of the source is automatically taken into account.

3. A Sound Authoring Tool

The Sound Authoring Tool is an ongoing development effort. At this time, the main structure of the program has been established and some working examples (e.g. see fig. 1) have been completed. While several sound design softwares, such as the visual environments *Kyma* or *Max/MSP*, are available off the shelf [4], most of them require a very sophisticated user, capable to conceive a sound model and to implement it by means of low-level building blocks. On the other hand, SAT is based on a somewhat rigid model architecture, but it provides intuitive controls without sacrificing much generality of conceivable sounds.

For the implementation of SAT we chose the Java language, since it is platform independent, distributed, and it has a compact windowing toolkit. Actually, one of the main reasons for using Java is the availability of a mechanism for arranging computations in asynchronous or synchronized threads. This mechanism has led to simple code by exploitation of the intrinsic modularity of the SAT architecture. Performance was a source of concern when we decided to use Java as a development language. Even though the interpreters which are currently available are fast enough for most purposes, we are aware of the fact that very critical numerical cores might be written in C language and embedded into native methods in order to allow real-time playback.

3.1. Modularity and Multithreading

Modularity is inherent in the architecture that we have proposed in sec. 2. In the software implementation, we tried to exploit modularity as much as possible. Therefore, we have conceived the various processing blocks as concurrent threads passing the signal samples to each other. A notable example is found in the implementation of the Exciter-Interaction-Resonator architecture. Here, Exciter and Resonator exchange their samples in a locked fashion by means of the Interaction's methods `get_res()`, `put_res()`, `get_exc()` and `put_exc()`. The Interconnection switches among a catalog of known `player.combinations`, corresponding to different instrument configurations, thus adapting the `get_*()` and `put_*()` methods to the actual configuration being used.

The Resonator communicates with the Exciter without talking to it directly and without knowing what the Exciter does. The Interconnection acts as a switchboard, being delegated to exchange data between the two peer objects Resonator and Exciter. The Interconnection, being the only place where the sound generator is composed and represented, is the only class that needs to be specialized to handle additional structures.

3.2. Control of Model Parameters

The problem of controlling sound processing algorithms is one of the most challenging issues raised to sound designers of the future [8]. Much of the timbre naturalness and "playability" depends on how the parameters are varied in time. Therefore, it makes sense to design models for control signals. Such models should be driven by attributes describing the expressive intentions associated with a given sound event, and therefore they should incorporate a model of expressivity. For controlling physical models, it would also be useful to adopt models of the dynamics of human gestures.

The models of control should be collected as a high-level software layer constructed above the parameters provided by the three layers of the SAT architecture. Extensions such as these will certainly be provided in future implementations, as soon as solid models of control will be available. At the present stage of development, there are three ways of controlling the parameters of sound objects represented in SAT:

- **Direct manipulation**. A sub-application, called the SoundHandler, provides a collection of sliders associated with the variables declared `handle`, and facilities for monitoring the waveforms (see fig. 3). In this way, real-time sound modification and parameter tuning are easily obtained. Interactive adjustment of parameters proved to be a key feature for rapid convergence towards satisfactory sound objects. The knowledge acquired at this stage can be used in the other modalities of control.

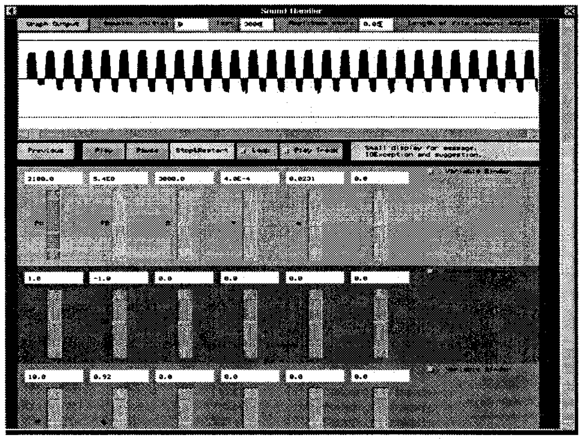


Figure 3: SoundHandler

- **Score**. A sub-application, called the TrackEdit, has been developed for designing complex trajectories in the space of parameters. These control "scores" can be played back within the SoundHandler and are run as a separate thread modifying the model parameters. In our implementation, score lines are intended as continuous streams of data rather than sequences of discrete events. This metaphor is closer to the preferred use of SAT as a monitor of software processes than to the traditional musical organization of sounds. The metaphor is not restrictive though, since "notes" can naturally result from the association of parameters with thresholds. For instance, the mouth pressure has to rise above a threshold in order to trigger a note in a reed instrument.
- **Remote calls**. The third way of controlling SAT is by means of socket connections. In Java, the `ServerSocket` class responds to remote requests of sound services (i.e. modifications of sound model parameters), thus providing extended accessibility to the sound objects which have been authored within SAT.

An aspect that is worth considering is that of the rate of control signals. These signals are usually derived form human gestures, thus having limited time variability. If we assume that the bandwidth of control signals is limited to, say, $100Hz$, we can use a rate much lower than the sampling rate for updating the parameters, thus saving many computations and reducing the inter-thread communication overhead. However, for certain parameters, e.g.

the length of delays, it is mandatory to smooth out abrupt transitions. It is widely recognized that this smoothing is better performed within the sound processing modules themselves [7], a viable strategy being insertion of simple first-order lowpass filters for those parameters which are considered to be sensitive to abrupt changes. This trick can be interpreted as a first-order approximation to the insertion of human-body dynamics within the models of the sounding objects.

4. Conclusion

We have introduced a new three-layer architecture for sound authoring tools, based on prior studies in perception and processing of sound. Physical models of sound sources, spectral models of sound, and physical/geometric models of spatial attributes are used. The large variety of possible sound sources is dealt with by composition of a limited number of basic sound-production mechanisms and control of several model parameters.

An actual Sound Authoring Tool has been implemented as a portable Java program, where threads are extensively used to exploit the modular nature of the general architecture. Sounds can be controlled in real time by means of sliders, as well as in a more controlled fashion with the aid of an editor of "parameter scores". Remote sonifications requests can also be received via a socket port.

In the future, we will enrich the SAT tool with several models of generation, modification, and ambience. Moreover, further efforts have to be devoted to the development of models for control signals. These models will form a high-level software interface for parameter manipulation, in such a way that both the quality and expressivity of sounds will be improved.

5. Acknowledgment

The TrackEdit has been developed by Marco Cortese.

References

[1] S. Barrass, "Sculpting a sound space with information properties," *Organised Sound*, vol. 1, no. 2, 1996.

[2] S. Barrass, *Auditory Information Design*. PhD thesis, Australian National University, 1997.

[3] D. R. Begault, *3-D Sound for Virtual Reality and Multimedia.* Boston, MA: Academic Press, 1994.

[4] N. Bernardini and D. Rocchesso, *Making sounds with numbers: a tutorial on music software dedicated to digital audio*, Proc. Workshop on Digital Audio Effects, DAFX98, Barcelona, Spain, Nov. 1998.

[5] J. Blauert, *Spatial Hearing: the Psychophysics of Human Sound Localization.* Cambridge, MA: MIT Press, 1983.

[6] G. Borin, G. De Poli, and A. Sarti, *Sound Synthesis by Dynamic Systems Interaction*, vol. Readings in Computer-Generated Music, pp. 139–160. IEEE Computer Society Press, 1992. D. Baggi, editor.

[7] R. B. Dannenberg and N. Thompson, "Real-time software synthesis on superscalar architectures," *Computer Music J.*, vol. 21, no. 3, pp. 83–94, 1997.

[8] G. De Poli, "In search of new sounds," *Computer Music J.*, vol. 20, pp. 39–43, Summer 1996. MIT Press.

[9] G. De Poli and P. Prandoni, "Sonological models for timbre characterization," *Journal of New Music Research*, vol. 26, no. 2, pp. 171–197, 1997.

[10] G. De Poli and D. Rocchesso, "Physically-based sound modeling," *Organised Sound*, vol. 3, no. 1, 1998.

[11] W. W. Gaver, *Using and Creating Auditory Icons*, vol. Auditory Display: Sonification, Audification, and Auditory Interfaces, pp. 417–446. Addison-Wesley, 1994. G. Kremer, editor.

[12] J. M. Grey, "Multidimensional perceptual scaling of musical timbres," *J. Acoustical Soc. of America*, vol. 61, no. 5, pp. 1270–1277, 1977.

[13] J. M. Hajda, R. A. Kendall, E. C. Carterette, and M. L. Harshberger, *Methodological Issues in Timbre Research*, vol. Perception and Cognition of Music, pp. 253–306. East Sussex, UK: Psychology Press, 1997. J. Deliege and J. Sloboda, editors.

[14] C. Hendrix and W. Barfield, "The sense of presence within auditory virtual environments," *Presence: Teleoperators and Virtual Environments*, vol. 5, no. 3, pp. 290–301, 1996.

[15] J.-M. Jot and O. Warusfel, "A Real-Time Spatial Sound Processor for Music and Virtual Reality Applications," in *Proc. International Computer Music Conference*, (Banff, Canada), pp. 294–295, ICMA, 1995.

[16] G. S. Kendall, "A 3-D Sound Primer: Directional Hearing and Stereo Reproduction," *Computer Music J.*, vol. 19, pp. 23–46, Winter 1995.

[17] G. S. Kendall, "The Decorrelation of Audio Signals and its Impact on Spatial Imagery," *Computer Music J.*, vol. 19, pp. 71–87, Winter 1995.

[18] G. Kramer, *Auditory Display: Sonification, Audification, and Auditory Interfaces.* Reading, MA: Addison-Wesley, 1994.

[19] C. Kyriakakis, "Fundamental and technological limitations of immersive audio systems," *Proc. IEEE*, vol. 86, pp. 941–951, May 1998.

[20] T. M. Madhyastha and D. R. Reed, "Data sonification: Do you see what i hear?," *IEEE Software*, vol. 12, no. 2, pp. 45–56, 1995.

[21] F. R. Moore, "A General Model for Spatial Processing of Sounds," *Computer Music J.*, vol. 7, no. 3, pp. 6–15, 1982.

[22] D. Rocchesso, "The Ball within the Box: a sound-processing metaphor," *Computer Music J.*, vol. 19, pp. 47–57, Winter 1995.

[23] D. Rocchesso and J. O. Smith, "Circulant and Elliptic Feedback Delay Networks for Artificial Reverberation," *IEEE Transactions on Speech and Audio Processing*, vol. 5, pp. 51–63, Jan. 1997.

[24] P. Schaeffer, *Traité des Objets Musicaux.* Paris, France: Éditions du Seuil, 1966.

[25] R. M. Schafer, *The Tuning of the World.* Toronto, Canada: McClelland and Stewart Limited, 1977.

[26] E. Zwicker and H. Fastl, *Psychoacoustics: Facts and Models.* Berlin, Germany: Springer Verlag, 1990.