3D-audio object oriented coding

Guillaume Potard
University of Wollongong

3D-AUDIO OBJECT ORIENTED CODING

By

Guillaume Potard

SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY
AT
UNIVERSITY OF WOLLONGONG
NORTHFIELDS AVE
WOLLONGONG NSW 2522
AUSTRALIA
SEPTEMBER 2006

UNIVERSITY OF WOLLONGONG
DEPARTMENT OF
SCHOOL OF ELECTRICAL, COMPUTER AND
TELECOMMUNICATIONS ENGINEERING

The undersigned hereby certify that they have read and recommend to the Faculty of Faculty of Informatics for acceptance a thesis entitled "**3D-audio object oriented coding**" by **Guillaume Potard** in partial fulfillment of the requirements for the degree of **Doctor of Philosophy**.

Dated: September 2006

External Examiner: _____

Research Supervisor: _____
Ian Burnett

Examing Committee: _____
Peter Svensson

_____
Stephen Barrass

UNIVERSITY OF WOLLONGONG

Date: **September 2006**

Author: **Guillaume Potard**

Title: **3D-audio object oriented coding**

Department: **School of Electrical, Computer and Telecommunications Engineering**

Degree: **Ph.D.** Convocation: **October** Year: **2006**

Permission is herewith granted to University of Wollongong to circulate and to have copied for non-commercial purposes, at its discretion, the above title upon the request of individuals or institutions.

Signature of Author

I would to thank my supervisor, Ian Burnett, who guided me through the years of the PhD with useful advice and encouragement.

I then dedicate this thesis to my partner, Megan Sproats, for all her support, patience and love.

Thank you to my parents for helping me to study in Australia and a big hello to my brother.

Table of Contents

List of Tables

List of Figures

ABSTRACT

This thesis first presents a novel object-oriented scheme which provides for extensive description of time-varying 3D audio scenes using XML. The scheme, named XML3DAUDIO, provides a new format for encoding and describing 3D audio scenes in an object oriented manner. Its creation was motivated by the fact that other 3D audio scene description formats are either too simplistic (VRML) and lacking in realism, or are too complex (MPEG-4 Advanced AudioBIFS) and, as a result, have not yet been fully implemented in available decoders and scene authoring tools. This thesis shows that the scene graph model, used by VRML and MPEG-4 AudioBIFS, leads to complex and inefficient 3D audio scene descriptions. This complexity is a result of the aggregation, in the scene graph model, of the scene content data and the scene temporal data. The resulting 3D audio scene descriptions, are in turn, difficult to re-author and significantly increase the complexity of 3D audio scene renderers. In contrast, XML3DAUDIO follows a new scene orchestra and score approach which allows the separation of the scene content data from the scene temporal data; this simplifies 3D audio scene descriptions and allows simpler 3D audio scene renderer implementations. In addition, the separation of the temporal and content data permits easier modification and re-authoring of 3D audio scenes. It is shown that XML3DAUDIO can be used as a new format for 3D audio scene rendering or can alternatively be used as a meta-data scheme for annotating 3D audio content.

Rendering and perception of the apparent extent of sound sources in 3D audio displays is then considered. Although perceptually important, the extent of sound sources is one the least studied auditory percepts and is often neglected in 3D audio displays. This research aims to improve the realism of rendered 3D audio scenes by reproducing the multidimensional extent exhibited by some natural sound sources (eg a beach front, a swarm of insects, wind blowing in trees etc). Usually, such broad

sound sources are treated as point sound sources in 3D audio displays, resulting in unrealistic rendered 3D audio scenes. A technique is introduced whereby, using several uncorrelated sound sources, the apparent extent of a sound source can be controlled in arbitrary ways. A new hypothesis is presented suggesting that, by placing uncorrelated sound sources in particular patterns, sound sources with apparent shapes can be obtained. This hypothesis and the perception of vertical and horizontal sound source extent are then evaluated in several psychoacoustic experiments. Results showed that, using this technique, subjects could perceive the horizontal extent of sound sources with high precision, differentiate horizontally from vertically extended sound sources and could identify the apparent shapes of sound sources above statistical chance. In the latter case, however, the results show identification less than 50 % of the time, and then only when noise signals were used. Some of these psychoacoustic experiments were carried out for the MPEG standardisation body with a view to adding sound source extent description capabilities to the MPEG-4 AudioBIFS standard; the resulting modifications have become part of the new capabilities in version 3 of AudioBIFS.

Lastly, this thesis presents the implementation of a novel real-time 3D audio rendering system known as CHESS (Configurable Hemispheric Environment for Spatialised Sound). Using a new signal processing architecture and a novel 16-speaker array, CHESS demonstrates the viability of rendering 3D audio scenes described with the XML3DAUDIO scheme. CHESS implements all 3D audio signal processing tasks required to render a 3D audio scene from its textual description; the definition of these techniques and the architecture of CHESS is extensible and can thus be used as a basis model for the implementation of future object oriented 3D audio rendering systems.

Thus, overall, this thesis presents contributions in three interwoven domains of 3D audio: 3D audio scene description, spatial psychoacoustics and 3D audio scene rendering.

Chapter 1

Introduction

1.1 3D audio object oriented coding and rendering

By nature, human hearing is spatial[1]. It is thus advantageous to exploit this ability to improve realism and immersion in virtual reality systems and cinema, increase the amount of information deliverable to users in sonification and alarm systems, improve the naturalness of teleconferencing systems or increase the listening pleasure of music. Three-dimensional (3D) audio or spatial audio is a set of technologies that exploits this spatial hearing ability by synthesising or preserving certain spatial auditory cues used by the brain to localise sound sources and to perceive their sizes, to perceive the spaciousness of reverberant environments etc. However, in contrast to traditional stereo sound reproduction, 3D audio technologies define new challenges in the recording, description, compression, transmission and rendering of 3D audio content.

Recently, with the creation of the VRML and MPEG-4 AudioBIFS standards, a new approach has been established to transmit 3D audio content: the object oriented approach. Instead of transmitting several fixed full bandwidth channels, this approach transmits the individual objects composing the scene as discrete entities. A scene description is also transmitted so that from this description and the scene

[1] Since we have two ears!

objects, the receiver is able to reconstruct the encoded 3D audio scene. This approach first improves scalability since less important objects of a scene can be discarded or replaced by lesser quality objects. Secondly, the transmitted 3D audio scenes can be altered at the decoder stage since objects are individually modifiable; this allows interactivity with the scene. Lastly, the object oriented approach lets content creators to author 3D audio scenes that are independent of a particular 3D audio speaker configuration or technology, since the 3D audio scenes are abstracted and rendered at the user terminal.

MPEG-4 and VRML, to describe audio and graphical scenes, rely on a scene graph model. While this approach is viable for describing interactive 3D graphical scenes, this thesis shows that it is inefficient when describing animated 3D audio scenes, because the scene graph model aggregates scene content and scene timing descriptions. In particular, describing complex scene animation can lead to complex and tangled scene graphs in VRML or MPEG-4 AudioBIFS, or requires an external mechanism to animate the scene (BIFS-anim for MPEG-4, no equivalent in VRML). After highlighting the issues related to the scene graph model, this thesis presents a novel 3D audio scene description scheme based on XML called XML3DAUDIO. The novel scheme is based on a new scene orchestra and score approach which simplifies and centralises the description of scene animation. This new approach, in turn, allows scenes to be easily re-authored and modified since the scene content description and the scene temporal description are individually accessible. XML3DAUDIO includes state of the art 3D audio description capabilities found in MPEG-4 Advanced AudioBIFS (AABIFS) and adds new features such as the description of the size of sound sources and allows algorithmic composition of 3D audio scene. XML3DAUDIO is compared against the scene graph model in several 3D audio scene examples. Although only applied to 3D audio scenes in this thesis, the new scene orchestra and score approach could also be used to describe 3D audio-visual scenes.

This thesis then presents novel developments on a technique which can be used to render the apparent extent and shape of sound sources in 3D audio displays. This technique relies on several decorrelated point sources which are spatialised at certain positions, so that, one and two dimensional broad sound sources can be devised. This technique also permits rendering sound sources with apparent shapes, by arranging the point sources in certain patterns. In 3D audio displays, rendering of sound source extent is an attractive feature since it contributes to higher realism and is used to devise broad sound sources such as wind blowing in trees, thunder etc. This thesis also proposes to use the apparent extent of sound sources to be used to convey information, that is, to be used in the context of data sonification. The technique for rendering sound source extent and its new developments are then subjectively tested in several novel psychoacoustic experiments where subjects had to identify and draw sound sources having particular extents and shapes. The results of these experiments give new insights into the perception of the apparent extent of sound sources by humans and provide recommendations and guidelines when rendering source extent in 3D audio displays. Some of these experiments were originally carried out by the author for the MPEG standardisation body to study the feasibility and need to add sound source extent description capabilities to the MPEG-4 AudioBIFS standard which could only represent point sound sources. It is then shown how the work and experiments presented in this thesis resulted in the creation of a new AudioBIFS node called *WideSound*, which now allows the description of spatially extended sound sources in MPEG-4 AudioBIFS scenes.

This thesis finally presents the implementation of a new real-time system for rendering 3D audio scenes. This system, called CHESS (Configurable Hemispheric Environment for Spatialised Sound), allows 3D audio scenes that are described with the new XML3DAUDIO scheme to be rendered and delivered to a small audience (3-4 people), using a 16-speaker array placed on a novel configurable scaffold. CHESS

follows a client-server architecture where the client performs XML parsing, management and update of the scene and where the server performs all the necessary signal processing tasks (spatialisation etc.). The architecture of CHESS and the 3D audio techniques that are employed are justified and compared with other techniques. CHESS provides a new model to implement subsequent 3D audio rendering systems. CHESS is a versatile system and was employed to carry the psychoacoustic experiments described in this thesis. The creative potential of CHESS is also highlighted in major projects where the system was used.

1.2 Thesis Outline

This thesis is organised as follows:

Chapter 2 reviews non-object and object oriented techniques for transmitting 3D audio scenes. The two approaches are compared. A review of the VRML and MPEG-4 AudioBIFS standards is then given. Problems associated with the scene graph model used by VRML and MPEG-4 AudioBIFS are highlighted. Spatial auditory perception is then reviewed. Emphasis is placed on the perception of the extent of sound sources in the case of one and multiple sound sources and in reverberant conditions. Several existing techniques for controlling the extent of sound sources are then critically reviewed and compared. Focus is then placed on a technique that uses multiple decorrelated sound sources to render the apparent extent of sound sources. Several signal decorrelation techniques are finally reviewed.

Chapter 3 presents XML3DAUDIO, a novel object-oriented XML scheme for describing 3D audio scenes. This technique follows the novel scene orchestra and score approach instead of the traditional scene graph model found in VRML and MPEG-4 AudioBIFS. It is highlighted that this technique simplifies the description of animated 3D audio scenes compared to the scene graph model, while featuring state of the art

3D audio description capabilities. An alternate use of the scheme as a meta-data annotation scheme for describing 3D audio content is also explained.

Chapter 4 presents a psychoacoustic study of the perception of apparent sound source extent and shape which are artificially rendered using several decorrelated sound sources. Experiments are first carried out with real decorrelated sound sources (i.e. speakers) so as to provide optimal conditions allowing source shape perception. Experiments are then repeated with spatialised decorrelated sound sources so as to study the feasibility or rendering source extent and shape in actual 3D audio displays. Other presented experiments study the gain in realism when using broad sound sources in 3D audio scenes and study perceptual effects of dynamic and time varying decorrelation. From these experiments, recommendations are given on rendering sound source extent and shape in 3D audio displays. The implementation of new sound source extent description capabilities in MPEG-4 AudioBIFS as a result of this work is finally described.

Chapter 5 presents the implementation of a novel 3D audio rendering system called CHESS (Configurable Hemispheric Environment for Spatialised Sound). The different 3D audio signal processing tasks used in CHESS are described. The reasoning and justification behind the choice of certain 3D audio techniques over others are highlighted. The rendering in CHESS of 3D audio scenes described with the new XML3DAUDIO scheme is then explained. An evaluation of CHESS is then given. Finally, major projects where CHESS was used are outlined.

Chapter 6 defines areas and topics where the research presented in this thesis could be furthered.

1.3 Contributions

The most important contributions of this thesis are listed below. They are listed in order of appearance with the related chapter number and publications.

- Developed a novel object oriented scheme for describing 3D audio scenes based on XML called XML3DAUDIO and compared it against the scene graph model (chapter 3) [PB04c, PB02d]

- Showed that XML3DAUDIO can alternatively be used for meta-data annotation of 3D audio content. (section 3.5) [PB04c]

- Proposed that the apparent shape of sound sources can be rendered using several decorrelated sound sources.(sections 4.5 and 4.6) [PB03, PS03, PS02, PSS02] (Prior-art proposed rendering only the width of sound sources using decorrelated sound sources (e.g. [Ken95])).

- Confirmed that horizontal sound source extent can be rendered using decorrelated point sources. Studied variations of perceived extent due to change of stimulus loudness and signal type. Found that the density of decorrelated point sources is one of the most important factor and leads to loss of binaural fusion for low density and source extent underestimation for high density. (section 4.3) [PSS03]

- Discovered that listeners can differentiate sound sources with horizontal, vertical and rectangular extents and that extent perception depends on sound localisation accuracy and thus depends on sound source position (section 4.4). Suggested that this ability has practical applications in the field of human-machine interfaces such as data sonification [PB04a]

- Discovered that listeners could, under certain conditions, identify the shape of sound sources significantly above statistical probability. Factors influencing

shape identifications are: signal type, sound localisation accuracy and apparent geometry of the sound source. (section 4.5 and 4.6) [PB03, PS03]

- Demonstrated that listeners preferred 3D audio scenes that contained broad sound sources over scenes that contained only point sources. (section 4.7) [PSS02]

- Discovered that dynamic decorrelation improved the naturalness of spatially extended sound sources but could lead to listening fatigue. (section 4.8) [PSS03]

- Confirmed that the time constant at which the binaural system performs inter-aural cross-correlation analysis is around 80 ms [PB04a] (section 4.9). This result is similar to [CPCS05].

- Initiated and contributed to the creation of a new MPEG-4 AudioBIFS node called *WideSound* to represent spatially extended sound sources in MPEG-4 AudioBIFS. (section 4.10) [PB03, PS03, PS02, PSS02, PSS03]

- Developed a novel real-time system called CHESS for rendering 3D audio scenes to a small audience. (chapter 5) [PI03, SG04]

- Developed a new configurable speaker array which allows quick configuration changes (section 5.2.2)

- Derived 4th order Ambisonics spatialisation encoding equations (section 5.4.1)

- Implemented 4th order Ambisonics encoding and decoding algorithms into Max/Msp objects and VST Plugins (section 5.4.2)

1.4 Publications

1.4.1 Conference papers

Potard, G. and Burnett, I. (2004), "Decorrelation techniques for the rendering of apparent source width in 3D audio displays", in **proceedings of the 2004 Digital Audio Effects Conference (DAFX2004)**, Naples, Italy, October 5-8 2004.

Potard, G. and Burnett, I. (2004) "Control and measurement of apparent sound source width and its applications to sonification and virtual auditory displays" in: **Proceedings of the 10th International Conference on Auditory Displays (ICAD2004)**, Sydney, Australia, 6-9 July 2004

Potard G. and Schiemer, G. (2004) "Sonification of the coherence matrix and power spectrum of EEG signals", accepted Sonification piece and paper to the Listening to the mind Listening Concert, Sydney Opera House, 8th July 2004. In **proceedings of ICAD2004**, Sydney, Australia, 6-9 July 2004

Potard, G. and Burnett, I. (2004)"An XML-based 3D audio scene metadata scheme" in: **Proceedings of the 25th Audio Engineering Society (AES) conference**, London, UK, 17-19 June 2004, pp 102-112

Potard, G. and Ingham, S. (2003)"Encoding 3D sound scenes and music in XML", in: **Proceedings of the International Computer Music Conference (ICMC2003)**, Singapore, October 2003

Potard, G. and Burnett, I. (2003) "A study on sound source apparent shape and wideness" in: **Proceedings of ICAD2003**, Boston, USA, 6-9 July 2003, pp 25-28

Potard, G. and Spille, J. (2003)"Study of Sound Source Shape and Wideness in

Virtual and Real Auditory Displays" in: **Proceedings of the Audio Engineering Society 114th convention**, Amsterdam, March 2003, Preprint 5766

Potard, G. and Burnett, I. (2002)"Using XML schemas to create and encode interactive 3-D audio scenes" in: **Proceedings of Distributed Communities on the Web (DCW2002)**, Sydney Australia, April 2002, Lecture notes in Computer Science, Springer-Verlag, pp 193-202

Co-authored papers

Barrass S., Whitelaw M., Potard G. (2006), "Listening to the mind listening", Media International Australia, Practice-led research, No 118 - February 2006.

Schiemer, G., Potard. G. et al (2004),"Configurable Hemisphere Environment for Spatialised Sound", in **proceeding of the Australasian Computer Music Conference (ACMC)**, Wellington, New Zealand, July 1-3 2004.

Mark F. O'Dwyer, Guillaume Potard, Ian Burnett (2004): "A 16-speaker 3D audio-visual display interface and control system", in **Proceedings of ICAD2004**, Sydney, Australia, 6-9 July 2004

1.4.2 MPEG meeting input papers

Guillaume Potard, Jens Spille, Jeongil Seo, "Report on sound source wideness 3rd Core Experiment", **MPEG meeting input paper m9457**, Pattaya, Thailand, March 2003.

Guillaume Potard, Jeongil Seo, Jens Spille, "Report of the second MPEG-4 AudioBIFS Sound Source Wideness Core Experiment", **MPEG meeting input paper m9171**, Awaji Is., Japan, December 2002.

Guillaume Potard, Jens Spille, "Report on MPEG-4 AudioBIFS Sound Source Wideness Core Experiment", **MPEG meeting input paper m8995**, Shanghai, China, October 2002.

Guillaume Potard, Ian Burnett, "Refined descriptors for 3D Audio DIA", **MPEG meeting input paper m8915**, Shanghai, China, October 2002.

Guillaume Potard, Ian Burnett, "Proposal on sound source wideness and shape in MPEG AudioBIFS", **MPEG meeting input paper m8533**, Klagenfurt, Austria, July 2002.

Guillaume Potard, Ian Burnett, "Proposal on the use of Digital Item and Digital Item Adaptation to transmit interactive 3D Audio content", **MPEG meeting input paper m8553**, Klagenfurt, Austria, July 2002.

1.4.3 MPEG meeting output papers

Guillaume Potard, Jeongil Seo, Jens Spille, "Workplan for Sound Source Wideness Core Experiment", **MPEG meeting output paper w5387**, Awaji Is., Japan, December 2002.

Guillaume Potard, Jens Spille, "Workplan for Core Experiment on MPEG-4 AudioBIFS Sound Source Wideness", **MPEG meeting output paper w5207**, Shanghai, China, October 2002.

Guillaume Potard, Jens Spille, "Workplan for MPEG-4 AudioBIFS Sound Source Wideness Core Experiment", **MPEG meeting output paper w5039**, Klagenfurt, Austria, July 2002.

Chapter 2

Encoding and perception of 3D audio

2.1 Introduction

This chapter first reviews the channel and object oriented approaches for transmitting 3D audio content; the two approaches are compared in terms of scalability and efficiency. Several channel and object oriented techniques for encoding and transmitting 3D audio content are then reviewed. The VRML and MPEG-4 AudioBIFS standards which can be used to transmit 3D audio content in an object oriented way are then reviewed (section 2.4). Issues related to the scene graph model used by VRML and MPEG-4 AudioBIFS are then highlighted. This chapter then reviews spatial auditory perception and related psychoacoustic phenomena. Well known and extensively studied spatial auditory percepts (eg localisation) are first reviewed (section 2.5). This review then focuses on one of the least studied spatial auditory percept being the perception of sound source extent (section 2.6). This chapter then reviews several techniques to render sound source extent in 3D audio scenes (section 2.11). Emphasis is then placed on a technique which uses several decorrelated sound sources to render source extent (section 2.12). Finally, several signal decorrelation techniques which can be used for the purpose of source extent rendering are reviewed (section 2.13).

2.2 Encoding of 3D audio scenes

In order to transmit 3D audio scenes over a telecommunications channel, two approaches exist. The first relies on encoding 3D audio scenes on a certain number of audio channels (ie at least two). The channel oriented approach is not object oriented since the objects composing a 3D audio scene are mixed into common audio channels and are consequently not individually accessible (Fig. 2.1). The second approach for encoding and transmitting 3D audio scenes is object oriented. In the object oriented approach, all objects present in the scene (eg sound sources) are located in separate data spaces and are transmitted (or stored) separately, along with a scene description (Fig. 2.2). The 3D audio scene description defines the content of the scene, its spatial structure (ie positions of objects, mechanical relationships etc.) and its temporal behaviour (i.e. object trajectories, playing times of sound sources etc.). More advanced features such as reverberation and sound reflections can also be included in the scene. At the user end, the user terminal is then responsible for rendering the 3D audio scene from the scene object resources (eg sound files) and the scene description.

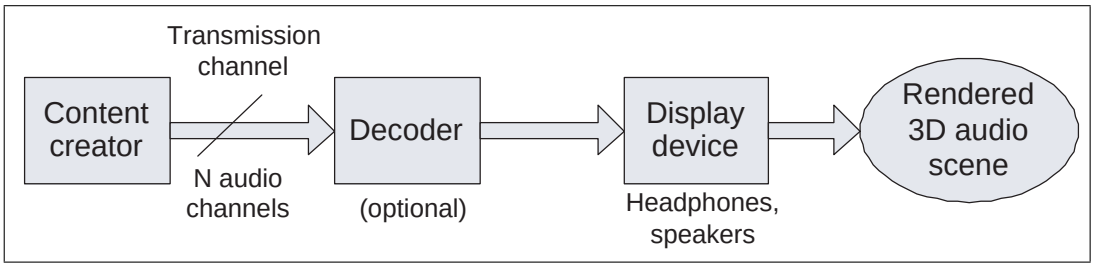


Figure 2.1: Transmission of 3D audio content using the channel oriented approach

Choosing between the channel and object oriented approach depends on the type of 3D audio scene to be transmitted, terminal capabilities and application. Firstly, three types of 3D audio scenes can be identified. There are first *natural* 3D audio scenes that are captured by 3D audio recording techniques (eg via binaural[1] or Ambisonics techniques[2]). There are then *synthetic* scenes that are artificially composed

[1] see 2.3.1
[2] see 2.3.3

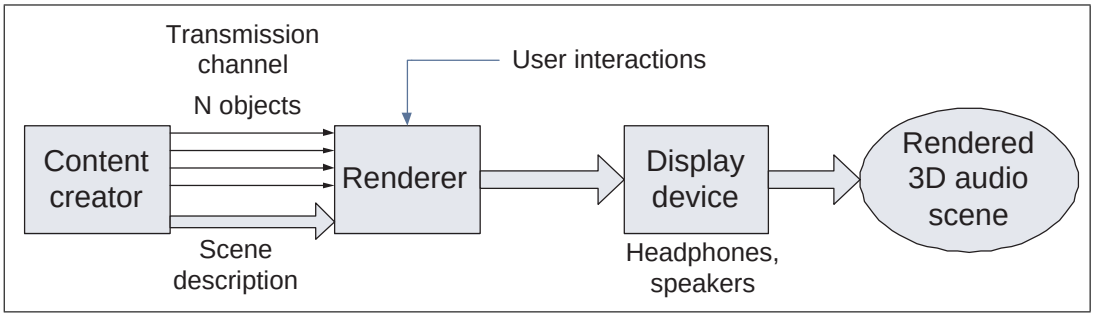


Figure 2.2: Transmission of 3D audio content using the object-oriented approach

by spatialising several sound sources and adding environmental factors such as reverberation. Finally, there are *hybrid* 3D audio scenes that can be devised by combining natural and synthetic 3D audio scenes. Object oriented encoding of 3D audio scenes is currently targeting synthetic and hybrid scenes since, for natural scenes, it is difficult to extract the audio content and properties (eg position etc.) of the individual objects present in the recording of a natural 3D audio scene. There is, however, ongoing efforts to achieve object extraction from 3D audio recordings in the research area of computational auditory scene analysis [Bre94].

The channel-oriented[3] approach, when encoding 3D audio content, has the advantage that the terminal is allowed to be relatively simple (i.e. simple playback is required), however, this approach has two main disadvantages: firstly, scalability is poor. The scalability problem implies that the end user must own the correct terminal configuration (eg correct number and placement of speakers). Secondly, the transmitted 3D audio scene is static and cannot be modified by the end user. If indeed, the scene does not need to be interactive (eg sound track of a movie) and that the user terminal configuration can be predicted (eg headphones or 5.1 surround speaker array[4]) then the channel oriented approach is suitable.

In contrast, the object oriented approach allows interaction and scalability of the scene. For instance, the same object-oriented 3D audio scene can be delivered to

[3]In the sense of one transmitted/stored audio channel per speaker; the Ambisonics technique reviewed in 2.3.3 does not fall into this category

[4]Defined by an ITU standard [ITU94]

a wide range of terminals (from a home theatre system to a mobile device) since the terminal is allowed to render and adapt the 3D audio scene according to its own rendering capabilities and in function of the display device (eg headphones or speakers). This adaptability mechanism simplifies content creation since only one version of the content is required. Another advantage of the object-oriented approach is that the 3D audio scene can be modified by the end user (i.e. interactivity) since individual objects and their parameters are available at the end terminal. This allows the object oriented approach to be suitable for interactive applications such as virtual reality, teleconferencing, sonification systems etc. A last advantage of the object-oriented approach is that individual sound objects of the scene can be encoded using optimal audio coders for their content type (i.e. speech or audio) and that synthetic audio content, requiring very little bandwidth, can also be included in object oriented 3D audio scenes.

One disadvantage of the object oriented approach, however, is that user terminals tend to be more complex and expensive since these are responsible for rendering 3D audio scenes from their descriptions. This issue can be resolved by the use of adaptation servers, which perform the required interpretation of the 3D audio scene descriptions and then stream the rendered 3D audio content to the terminal as audio channels. This technique is likely to grow in scope thanks to the recent creation of the MPEG-21 standard [BVdWH$^+$03, BGP03, Vet04].

Several channel oriented techniques for transmitting 3D audio content are now reviewed. These are followed by the review of several object oriented standards for encoding and transmitting 3D audio content (section 2.4).

2.3 Channel oriented encoding of 3D audio scenes

2.3.1 Binaural recording

Binaural recording requires only a normal stereo audio channel to transmit a full 3D audio sound field, thus it has the advantage of being compatible with existing audio formats such as the Compact Disk and FM radio. Based on psychoacoustics, binaural recording preserves the necessary spatial cues used by the brain to perform localisation of sound sources, size estimation of sound sources and other percepts such as source distance and environment perception[5]. Binaural recordings can simply be obtained by placing microphones in the ear canals of a subject or by using a dummy head microphone (Fig. 2.3). Binaural recordings then need to be played on headphones or two speakers using transaural cross-talk cancellation techniques [Bau93, JLW95].

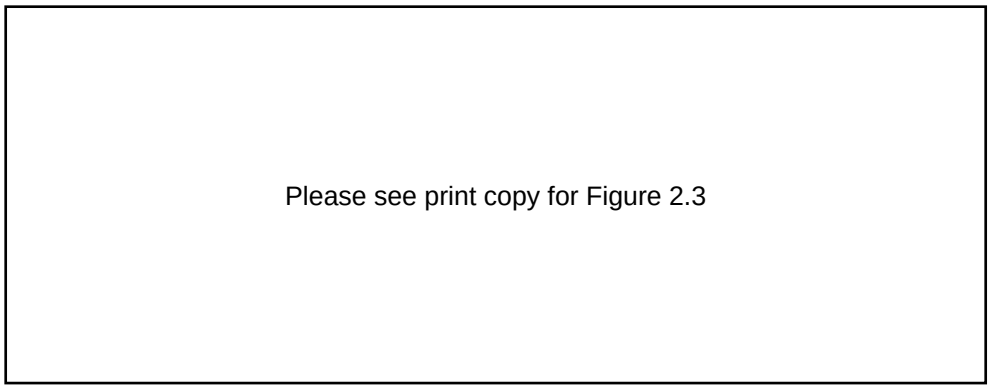


Figure 2.3: Dummy head microphone example for recording 3D audio scenes binaurally (Neumman KU100 model)

Binaural recordings can also be artificially created by convolving a monaural audio file with a Head-Related Transfer Function (HRTF) filter database [Beg92a, JLW95]. One drawback of binaural recordings, however, is that head rotations of the user

[5]These spatial auditory percepts are reviewed in section 2.5

should be restricted since the virtual sound field rotates with the user's head; this tends to create confusion in front/back source localisation and is unnatural [Beg94]. A solution to this problem is to use a head-tracking device that records the real-time orientation of the listener head; from this information, a binaural filter is selected or interpolated from the HRTF filter database [ADT04] and convolved with the sound source signal. Although achieving high quality spatialisation, head tracked auralisation imposes a complex and low-latency terminal. In addition, 3D audio scenes cannot be prepared in advance due the unpredictability of the listener head orientation and thus each sound source must be stored separately; the object-oriented scheme presented in chapter 3 could be used to provide the necessary meta-data to transmit such scenes on a communication channel.

Binaural recordings also suffer from non-individualisation of the HRTFs, resulting in a mismatch between the HRTFs used during the binaural recording and the user's particular HRTFs. This results in poorer localisation, front/back confusions and non-externalisation of the auditory events [Beg91a].

2.3.2 Multi-channel techniques

ITU 5.1 surround

Other techniques use several full bandwidth audio channels to transmit 3D audio scenes [Dav03]. Each transmitted channel is rendered on a separate speaker of a given speaker array. These techniques are used in DVD and cinema applications (eg Dolby Surround [Dol98]). One of the most widely spread surround sound format being the 5.1 ITU norm [ITU94, Bos00] which is encoded in Dolby AC3 format [TDD$^+$94] on DVDs. The ITU 5.1 speaker placement norm is depicted in Fig. 2.4.

With multi-channel 3D audio techniques, amplitude panning [Wes98, Pul97] is commonly performed between pairs of channels to create the illusion of virtual sound sources that can be placed around the central listener. Other spatialisation techniques have also been used in the 5.1 surround context, such as Ambisonics [Ger92a, Ger92c]

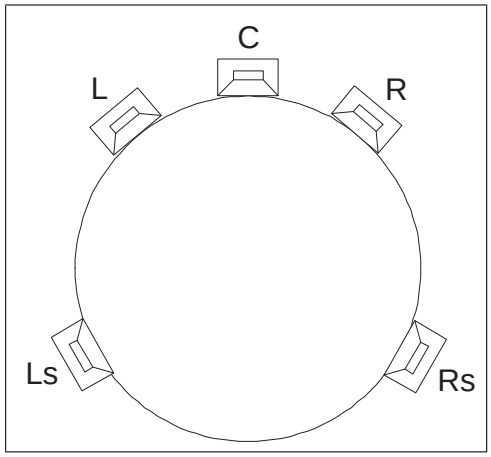


Figure 2.4: 5.1 Surround speaker positioning as defined by the ITU BS.775-1 recommendation

and Ambiophonics [Gas03]. Using psychoacoustic concepts, these techniques aim at improving spatialisation accuracy and increase the sweet spot area compared to basic amplitude panning.

In terms of efficiency, one drawback of the 'channel per speaker' approach is that the required number of channels is relatively high and directly equal to the number of speakers used. Another drawback of this approach is that the configuration of the speakers remains fixed, forcing the user to use the correct speaker configuration, restricting flexibility and scalability of the format (it is, however, possible to render 5.1 surround content on headphones using binaural spatialisation [Kyr00, Lak]).

In order to decrease the bitrate, traditional approaches of audio coding are often applied to multi-channel 3D audio content whereby each individual channel is compressed using perceptual concepts (eg MPEG-2 layer 3). However, the psychoacoustic models used for compressing the individual channels are often based on monaural psychoacoustic models. This can result in noise-masking models that, in a multi-channel context, can add to perceptible quantisation noise levels when these channels are played back at different positions in space [Spi03].

Another weakness of compressing channels individually, is that spatial masking concepts and inter-channel redundancy are not used to reduce the bitrate. This

issue is being addressed, and current research focuses on developing a general spatial masking model. Other researchers in the MPEG standardisation body are currently using these concepts to implement what is known as spatial audio coding [JvSBC03] and binaural cue coding [Fal04].

Without using psychoacoustic concepts, certain multi-channel techniques (eg Dolby surround [Dol98]) reduce the number of transmitted channels by matrixing the audio channels onto a smaller number of channels. In the case of Dolby Surround, four channels are matrixed into two. At the decoding stage, de-matrixing is then used to obtain the original speaker signals[6]. During this process, cross-talk between channels occurs, leading to a smeared 3D audio impression, poorer localisation and limited audio bandwidth.

It should be noted that the 5.1 format is not based on psychoacoustic theory, but is instead dictated by cinema and home entertainment industry requirements. As a result, 5.1 surround can only reproduce horizontal (2D) audio scenes and due to the non-uniformity of the speaker array, localisation blur and sound source instability occur on the listener's sides. Improvements have been proposed, using more side speakers, such as the 7.1 format. Despite some improvements, this technique still remains inferior to real 3D audio techniques such as Ambisonics which are based on solid mathematical foundations (see section 5.4.1).

Arbitrary configurations

In theory, it is possible to transmit 3D audio content using an arbitrary number of channels and an arbitrary speaker configuration. Currently, however, none of these configurations are standardised. Since rendering 3D audio scenes on speakers[7] requires at least eight channels (cubic configuration) for hemispheric reproduction, transmitting 3D audio content using a 'channel-per-speaker' method is highly inefficient and

[6] This is why Dolby is said to use 4:2:4 matrixing

[7] Using non transaural techniques

best avoided. Ambisonics and object-oriented approaches which are now reviewed remain the only viable solutions for transmitting high quality 3D audio content flexibly and efficiently.

2.3.3 Ambisonics

Ambisonics is a 3D audio multichannel technique invented in the 70s [Ger75] which has recently been improved to higher orders [Dan03a]. In contrast to multi-channel techniques described in the previous section, Ambisonics encodes a complete 3D audio scene onto a finite number of channels which are known as the B-format. Ambisonics, in its first order form, encodes a 3D audio scene onto four audio channels [Mal95]. First order B-format encodes, for one point in space, the omnidirectional sound pressure (W) and sound velocities in the three directions of space (X,Y,Z). First order B-format directivity patterns are depicted in Fig. 2.5. At the user terminal stage, the B-format is then decoded or de-matrixed to match the user particular speaker configuration (Fig. 2.6).

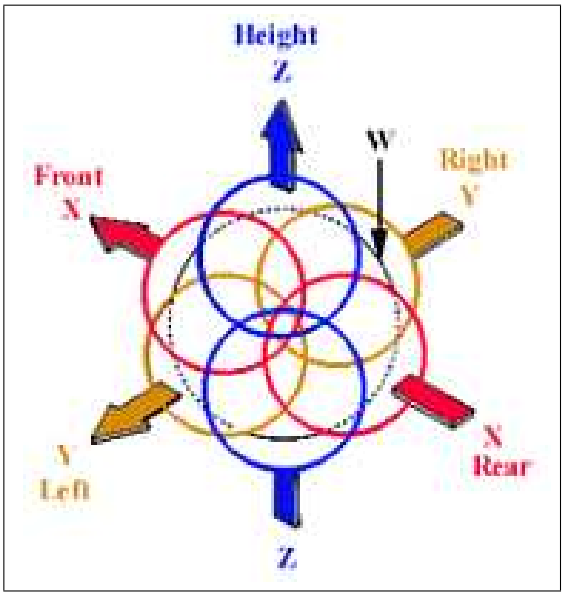


Figure 2.5: Schematic view of the B-format W,X,Y and Z channel directivity patterns

The Ambisonics approach has the advantage that the encoded B-format 3D audio content remains independent of a particular target speaker configuration; this results in a higher flexibility and versatility over channel-per-speaker techniques. For irregular speaker arrays, however, it is not always possible to compute the respective decoding matrix and thus, Ambisonics is best suited for regular speaker configurations, such as the circular, cubic or geodesic dome[8] configurations.

With higher order Ambisonics, the description of the sound field at one point requires a higher number of encoding channels[9] but a more detailed description of the sound field is obtained [Mal99b]. As a result, higher order Ambisonics improves spatialisation accuracy and image sharpness [Dan00]. Higher order Ambisonics is detailed in section 5.4.1.

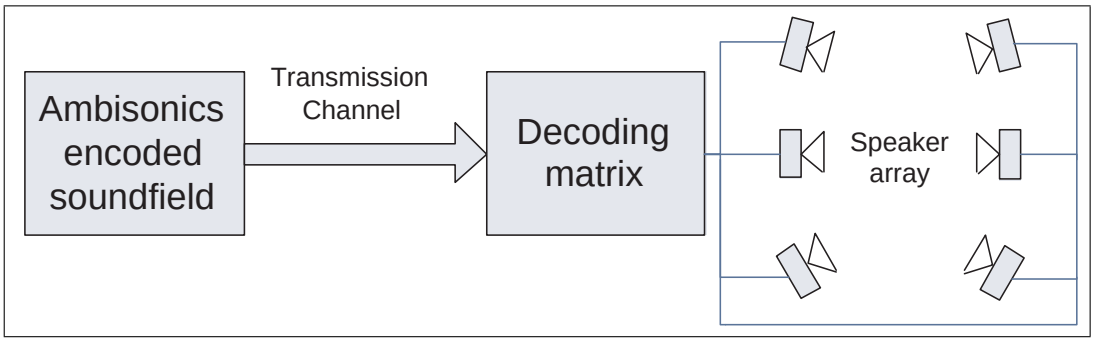


Figure 2.6: Overview of the Ambisonics encoding/decoding approach

Ambisonics can be used to capture and encode natural 3D audio scenes in first order B-format using a special Soundfield microphone (Fig. 2.7). Recent research developments highlighted the possibility of recording 3D audio scenes using higher order Ambisonics microphones [Lab03].

Alternatively, Ambisonics can be used as a spatialisation algorithm to create synthetic 3D audio scenes (section 5.4.1). This is achieved by matrixing a monaural signal into an artificial B-format signal. The B-format encoded signal is then decoded to the target speaker configuration. Ambisonics also allows hybrid 3D audio

[8]The geodesic dome is the chosen configuration for the 3D audio rendering system described in section 5 of this thesis, see 5.4.1

[9]Exactly $(n+1)^2$ channels where n is the Ambisonics order

scenes to be devised, whereby, a natural B-format recording is mixed with artificially created B-format signals. Ambisonics can thus be used to encode natural, synthetic and hybrid 3D audio scenes.

Figure 2.7: Tetrahedral configuration of capsules inside the Soundfield microphone

A common problem with Ambisonics and other spatialisation techniques[10] is that the rendered 3D audio field is correctly reproduced on the condition that the user is located in the 'sweet spot' area, placed at the centre of the speaker array. It has been formally demonstrated [Dan00] that this problem can be solved by using higher order Ambisonics (HOA), which increases the sweet spot area as well as localisation sharpness and accuracy. A mathematical formulation of Ambisonics and HOA is given in section 5.4.1.

There have been ongoing efforts to adapt B-format content to traditional stereo transmission mediums, this is the case of the UHJ format [Ger85] which, by using a matrixing technique, can encode two-dimensional B-format[11] content onto two audio channels. Due to cross-channel leaking issues, however, the UHJ format remains

[10] Except the Wave Field Synthesis technique [VB99]

[11] Which requires three audio channels

inferior to the B-format in terms of spatialisation accuracy and stability.

Finally, it was shown [Ger92a] that it is possible to decode B-format content to a standard 5.1 setup using what is known as a Vienna decoder. This allows Ambisonics to be used in DVD and cinema applications. Compared to the normal use of amplitude panning in 5.1 systems, the use of Ambisonics improves spatialisation accuracy and user immersion [Dan00].

Effects of audio compression on B-format content have not been perceptually studied, however, it was mathematically demonstrated [Spi03] that quantisation noise levels would become audible when compressing B-format with traditional audio coders. Phase mismatch between channels caused by audio compression are likely to degrade the rendering quality of 3D audio content in terms of localisation accuracy and stability. In the case of Ambisonics, it is thus advisable to compress sound sources individually in a monaural or stereophonic form and to perform B-format encoding and decoding (i.e. the spatialisation) only at the user terminal stage. This is only achievable with the object oriented approach to 3D audio coding, which is now reviewed.

2.4 Object oriented encoding of 3D audio scenes

2.4.1 VRML and X3D

The Virtual Reality Markup Language (VRML) is an International Standard Organisation standard (ISO 14772-1) [VRM98] that is used to create and transmit[12] interactive virtual reality environments composed of 3D graphical and audio objects. Even though VRML is used to build audio-visual scenes, the present review is oriented towards 3D audio scene encoding and rendering. This review relates to VRML version 2.0 which is also known as VRML97. X3D [X3D] which is developed by the W3C consortium is the new version of VRML. X3D is the translation of the VRML

[12] typically through a web browser interface

standard into XML syntax with some added functionalities but, in terms of 3D audio scene description capabilities, has currently only the same basic capabilities of VRML.

Transmission model

In VRML, 3D audio-visual scenes are described in an ASCII[13] text file which semantically describes the scene, this file is known as a scene graph [ANM97]. The scene graph describes the objects present in the scene, eventual hierarchical relationships between them and the temporal and interactive behaviour of the scene. However, the scene graph model is oriented towards 3D graphical scene representation and it is shown in chapter 3 that the scene graph approach is not well suited when describing animated 3D audio scenes.

To render a scene, the VRML client first starts downloading the scene graph. After parsing the scene, the client then follows resource URLs, downloads and stores them in local memory (Fig. 2.8). VRML only supports uncompressed WAV and general MIDI files. A drawback of VRML is that the scene can only be rendered when all resources have been downloaded and stored in the local client memory, creating transmission delays for scenes where a great quantity of audio resources are used. X3D and MPEG-4, in contrast, are able to accept incoming audio streams, which reduces memory usage at the terminal and results in quicker rendering start.

After obtaining a copy of the scene graph and audio resources in local memory, the scene manager then parses the scene graph and sends commands to the rendering engine to instantiate sound objects of the scene (Fig. 2.8). The rendering engine then performs the necessary spatialisation and signal processing tasks to render the 3D audio scene. The scene manager is responsible for updating the scene during scene animation and user interaction and then sends appropriate commands to the rendering engine. The rendering engine can take the form of a software only implementation or can be hardware accelerated using 3D Audio APIs implemented for certain sound

[13] And in XML for X3D

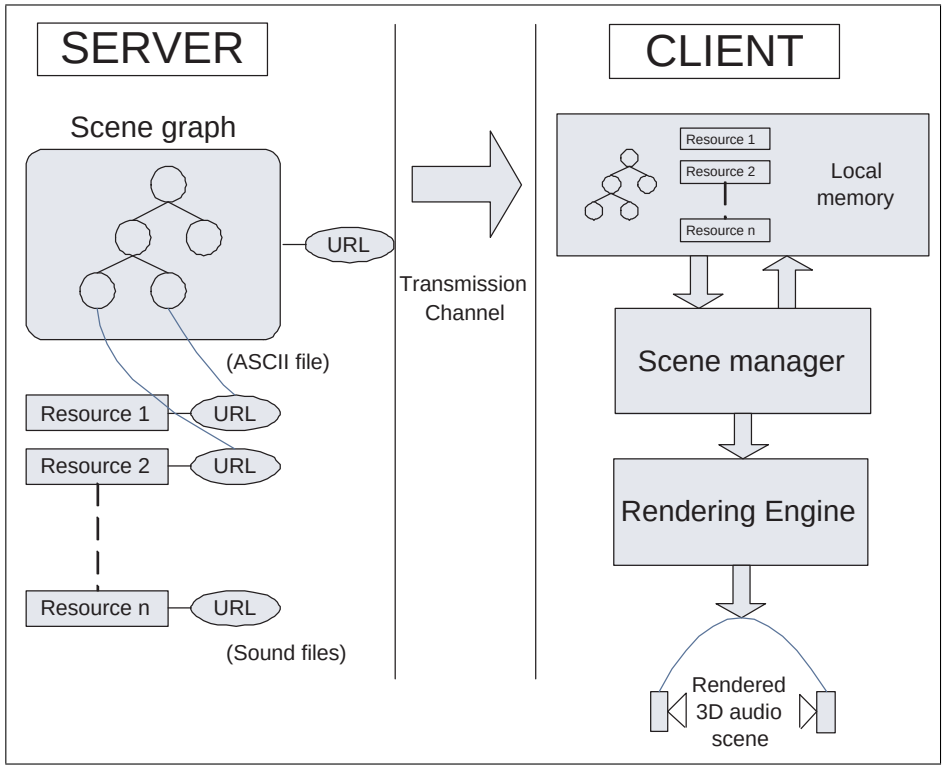


Figure 2.8: Overview of the VRML server/client architecture

cards.

Details of spatialisation and signal processing tasks are not part of the VRML standard. It is thus the responsibility of the rendering engine to perform the appropriate rendering in accordance with the user terminal configuration (eg headphones or speakers, selected spatialisation algorithm etc.). In contrast to the channel-oriented approach, the object oriented approach thus offers a greater versatility since the same scene can be transmitted to very different terminals without adaptation.

The scene graph model

The scene graph model follows a reversed tree architecture where the scene root has dependent child nodes which, in turn, can be parents of other nodes (Fig. 2.9). This model is used to reflect mechanical relationships between objects so that, when a

parent object is modified (for instance, moved), its children are automatically affected by the same transformation. To do so, transformation nodes are used to group and alter objects in a particular sub-branch of the scene graph (for example to perform rotation on a group of objects). The scene graph model also allows defining *routes* between objects so that complex and interactive scene behaviours are achieved. For example, a node detecting a mouse click can route an event to trigger the playing of a sound in an *AudioClip* node. In VRML, objects have a particular syntax with parameter fields. Each parameter field has a name, a data type, a default value and an exposed status. The former defines whether a particular data field can be externally modified by other objects using the parameter routing mechanism.

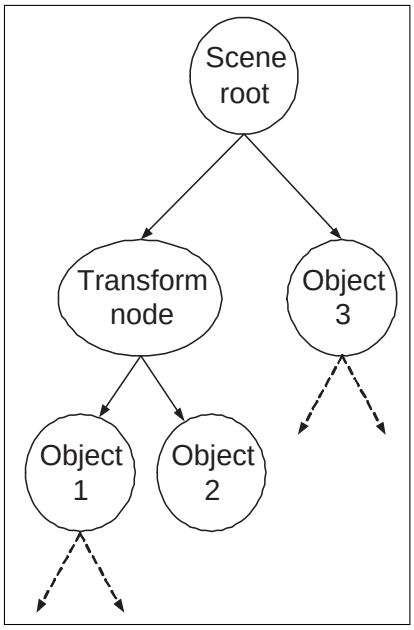


Figure 2.9: Schematic view of a scene graph

3D audio scenes in VRML

Objects which can be used to create 3D audio scenes in VRML are the *Sound*, *MovieTexture* and *AudioClip* nodes. The semantics of the *AudioClip* and *Sound* nodes are shown in Fig. 2.10. The *AudioClip* node is used to access audio resources (WAV or

MIDI) and the *Sound* node is used to spatialise the audio material in the scene. The *AudioClip* node is therefore a child of the *Sound* node. The *MovieTexture* node can be used instead of the *AudioClip* node when video content is also required.

In the *Sound* node, the position, orientation and directivity of the spatialised sound sources can be defined. The sound source directivity model of VRML is coarse (Fig. 2.11) and cannot describe accurate directivity patterns and frequency dependent directivity. In addition, VRML can only describe point sound sources which have no spatial dimensions; this results in poor realism of 3D audio VRML scenes (chapter 4). In terms of describing the acoustical properties of 3D audio scenes, VRML can describe geometrical shapes which can be used by the scene renderer to calculate reflections, occlusions, diffusion etc. However, since there is no provision in the VRML standard to describe material reflectivity properties, only simple reflection attenuation coefficients can be used. This is simplistic since, in reality, frequency and angle of incidence attenuation occurs during reflections of sound waves on surfaces [HSK97]. In addition, VRML lacks description capabilities to define the acoustics and reverberation of environments.

Despite some efforts [Ell98, TS01], VRML capabilities are thus insufficient for describing high quality 3D audio scenes. This is normal since VRML was originally designed as a visual virtual reality language. In section 2.4.2, it is shown how MPEG-4 AudioBIFS addresses these shortcomings.

Problems associated with the scene graph model

To perform scene animation, VRML defines *animation circuits* (Fig. 2.12). An animation circuit is devised by *TimeSensor* nodes acting as clocks and *PositionInterpolator* nodes containing a list of key frame values to interpolate. The animation circuit is complete by routing interpolated values to the position field of the object to animate. The duration, start and stop time of the animation are defined in the *start* and *stop* fields of the *TimeSensor* node and the animation path or trajectory is defined by the key frame values specified in the *keyframe* field of *Interpolator* node.

AudioClip {

Field Type	*Data Type*	*Field name*	*Default value*
exposedField	MFString	**url**	[ ]
exposedField	SFString	**duration**	“”
exposedField	SFTime	**startTime**	0.0
exposedField	SFTime	**spotTime**	0.0
exposedField	SFFloat	**pitch**	1.0
exposedField	SFBool	**loop**	FALSE
eventOut	SFBool	**isActive**	
eventOut	SFFloat	**duration _changed**	

}

Sound {

Field Type	*Data Type*	*Field name*	*Default value*
exposedField	SFNode	**source**	NULL
exposedField	SFFloat	**intensity**	1.0
exposedField	SFVec3f	**location**	0.0 0.0 0.0
exposedField	SFVec3f	**direction**	0.0 0.0 1.0
exposedField	SFFloat	**minFront**	1.0
exposedField	SFFloat	**minBack**	1.0
exposedField	SFFloat	**maxFront**	10.0
exposedField	SFFloat	**maxBack**	10.0
exposedField	SFFloat	**Priority**	0.0
Field	SFBool	**spat ialize**	TRUE

}

Figure 2.10: Semantics of the VRML sound nodes

Animation in VRML can require a large number of *TimeSensor* and *Interpolator* nodes and routes if the animation of the scene is intricate. It was estimated by Walsh [WBS02] that scene temporal description may exceed 90 % of the scene description data for complex scene animation described over a long period of time. Furthermore, this increase in complexity is directly reflected in the syntactic structure of the scene description. The description of scene animation with the scene graph model can thus lead to a high level of scene graph structural complexity and creates hard-wired relationships in the scene graph. Hence, the non-separation of temporal (i.e. animation) and structural data (i.e. the objects) tends to create complex and tangled scene graphs which are poorly re-usable and inefficient.

Another issue with the scene graph model is that the *play* and *stop* times of sound sources are embedded in the fields of the *AudioClip* nodes themselves (Fig. 2.12). Therefore, if a sound source is played several times in a scene, several *AudioClip* nodes are required, resulting in description redundancy. Alternatively, one *AudioClip* node and an animation circuit can be used to modify the *start* and *stop* fields of the

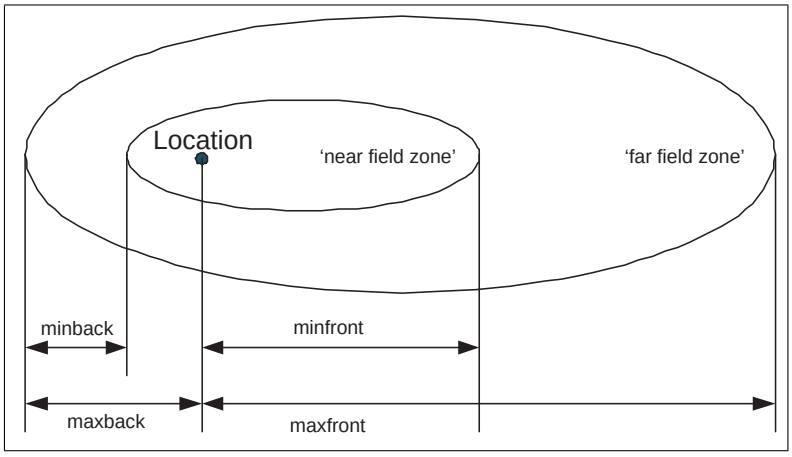


Figure 2.11: VRML sound source ellipsoidal directivity model with only four parameters

AudioClip node at certain times in the scene; this solution, again, increases scene graph complexity. The scene graph model thus results in non-centralised temporal information which is spread out across the whole scene graph. Besides, the scene temporal information may be deeply nested within objects of the scene. This, in turn increases the complexity required to access and process this data.

When describing animated 3D audio scenes, the scene graph model can thus be regarded as inefficient and results in high syntactic complexity. The decentralisation of the scene temporal behaviour description also prevents from easily re-authoring scenes with different timings. To overcome these issues when composing 3D audio scenes, this thesis presents a novel method for describing 3D audio scenes (chapter 3). This method does not use a scene graph approach and thus does not suffer from the issues that have been identified. This, in turn, provides a viable and efficient format for describing and rendering 3D audio scenes.

2.4.2 MPEG-4

MPEG-4, obtained the status of an ISO/IEC international standard [MPE99] in 1999 and diverges from the conventional approach to audio and video compression used

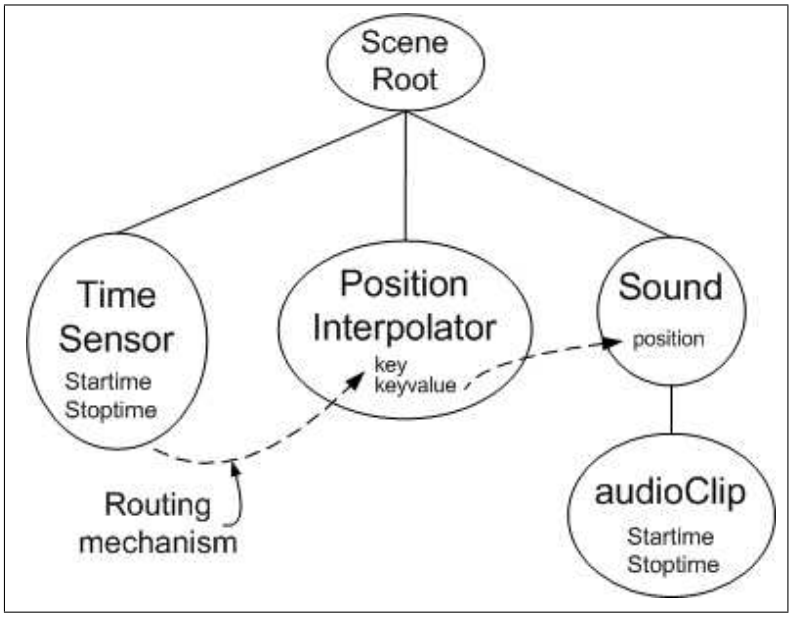


Figure 2.12: Illustration of an animation circuit in VRML

in MPEG-1 and MPEG-2. Following an object-oriented approach, MPEG-4 can accommodate a wide range of multi-media and virtual reality applications. With a standardised and efficient system layer [AEH+00], MPEG-4 incorporates new audio and video coding tools in addition to the previous MPEG codecs. A major improvement in MPEG-4 is the use of synthetic audio coders that can encode synthetic audio and speech with streams of parameters. In MPEG-4, natural and synthetic content can then be mixed in scenes.

Using streaming mechanisms, MPEG-4 scenes and media can be streamed, unlike VRML scenes, which must first be completely downloaded before rendering can be performed. A major difference with VRML is that the MPEG-4 standard encompasses the standardisation of the system layers; namely the delivery, synchronisation and compression layers (Fig. 2.13). These layers perform decoding of the raw MPEG-4 bit stream and demultiplex individual Elementary streams (ES) which can be either audio, video or control data. These streams are then synchronised to adjust for different random delays occurring during transmission. Finally, the individual ESs are decoded and used in MPEG-4 scenes using special nodes as entry points. This

architecture provides a complete end-to-end standardised solution for transmitting audio-visual and 3D audio scenes. Like VRML, MPEG-4 does not standardise details of the scene rendering engine, such as spatialisation and DSP effects (in the case of 3D audio) and thus MPEG-4 scenes can be delivered to terminals having different speaker configurations etc.

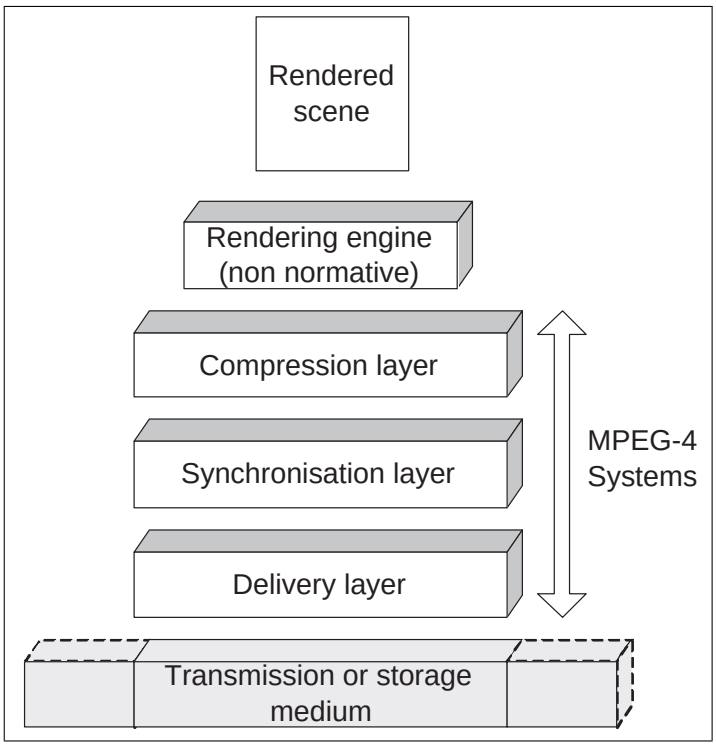


Figure 2.13: Standardised MPEG-4 system layers between raw bitstream and renderer

BIFS

To describe object oriented scenes, MPEG-4 uses an object oriented scene description mechanism: BIFS (BInary Format for Scenes) [MPE99, SVH99, SFE00]. A scene in MPEG-4 BIFS may range from a simple 2D web page to full 3D audio and visual immersive environments. A BIFS scene example is depicted in Fig. 2.14.

MPEG-4 BIFS has been built over VRML technology and thus follows a scene

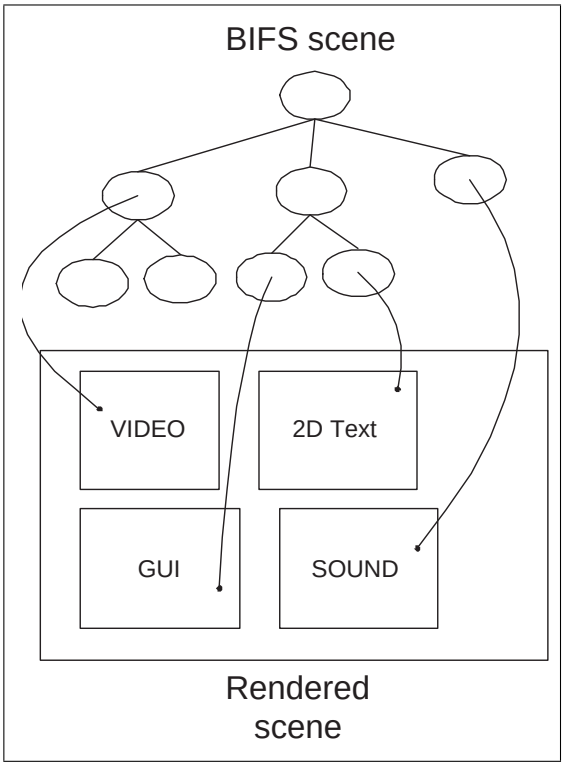


Figure 2.14: Example of BIFS scene containing video, audio, text and a graphical user interface

graph approach for representing audio-visual scenes. The nodes, routing and animation mechanisms of VRML can be found in MPEG-4 BIFS, and new nodes are exclusive to MPEG-4 BIFS. One main difference between MPEG-4 and VRML is that MPEG-4 uses a binary scene graph format, while VRML uses ASCII text. This results in better efficiency and smaller file size. An XML text version of MPEG-4 BIFS also exists: MPEG-4 XMT, which was created to improve cross-operability of MPEG-4 with X3D [X3D], SMIL [SMI] while preserving the scene author intentions. Two versions of XMT exist: XMT-A and XMT-Ω. While XMT-A is a straightforward translation of binary BIFS scenes to textual XML, XMT-Ω intends to provide a higher level of abstraction defining the author intentions. However XMT-Ω is based on SMIL which, in terms of describing 3D audio scenes, can be problematic (see section 2.4.3).

It was explained in section 2.4.1 that, in the scene graph approach, the scene objects that define the scene content, its temporal behaviour and its interactive behaviour are aggregated and are complexly linked with routes. This, in turn, results in high semantic complexity and poor re-usability of the scene description. To overcome these issues, MPEG-4 BIFS offers two additional scene animation and update mechanisms: BIFS-Commands and BIFS-Anim [WBS02, PE02]. After a scene has been transmitted and is being rendered, BIFS-Commands can be issued to perform single modifications of the BIFS scene, such as inserting or deleting a new object or changing the properties of a particular object in the scene. BIFS-Commands can be used, for example, to describe the sequencing of the sound sources by changing the *start* and *stop* times of *AudioSource* nodes at certain times in the scene. BIFS-Anim on the other hand, is used to generate continuous scene update commands encoded in separate binary streams. A BIFS-Anim stream can be used in 3D audio scenes, for instance, to animate sound sources along predetermined trajectories. BIFS-Commands and BIFS-anim binary streams are embedded in the general MPEG-4 bit stream which also contains media data; this is illustrated in Fig. 2.15.

Compared to intrinsic scene animation, that is, animation that is described in the scene itself using *TimeSensor* nodes, *Interpolator* nodes and routing, these two additional scene update features greatly simplify the description of scenes since a static scene content may be described and then animated externally. The novel 3D audio scene description scheme proposed in chapter 3, however, achieves the same simplification without requiring such additional scene animation mechanisms.

AudioBIFS

Emphasis is now placed on 3D audio scenes. In MPEG-4, these can be devised using AudioBIFS which is simply an audio subset of BIFS nodes. While VRML could only use downloadable Audio clips, MPEG-4 AudioBIFS scenes, with the help of the *AudioSource* node, can accept any audio streams coming from the system layers. These streams can be either natural encoded audio, or synthetic audio produced by

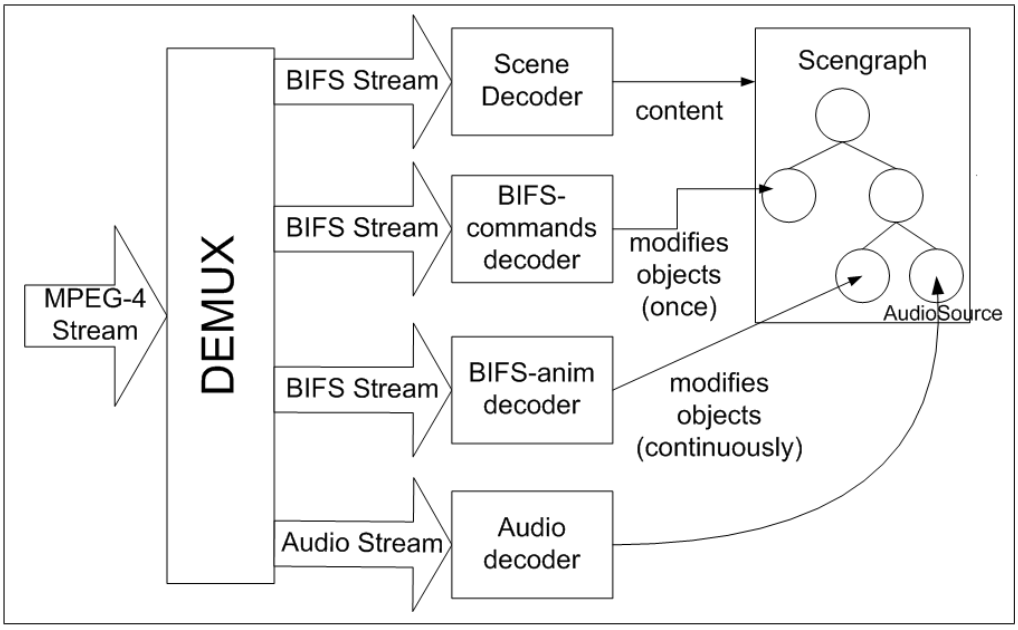


Figure 2.15: Illustration of BIFS-Commands and BIFS-Anim streams animating modifying the state of the scene graph is a timely manner

local sound synthesisers. These streams can then be further mixed, delayed, and processed using AudioBIFS nodes. Final audio streams are then spatialised in the audio scene using the *Sound* node. These three stages are shown in Fig. 2.16. The AudioBIFS nodes are listed in table 2.1.

In contrast to the VRML *Sound* node which could only accept *AudioClip* nodes as children, the *Sound* node in AudioBIFS can accept any audio streams, directly from the output of an audio decoder or from the result of audio processing and mixing tasks performed by other AudioBIFS node. Therefore, the *Sound* node is always located at the topmost of an audio subtree. In version 1 of AudioBIFS, however, 3D audio capabilities are similar to that of VRML, that is, a simple ellipsoidal sound source directivity model is used and no reverberation or acoustical properties could be defined, resulting in poor 3D audio rendering capabilities and realism. This motivated 3D audio experts at MPEG to create Advanced AudioBIFS (AABIFS) which, finally, provides state of the art 3D audio description capabilities.

Node name	Function
Sound	(VRML inherited) Spatialises a stream inside a 3D audio scene
AudioClip	(VRML inherited) Provides an entry point into BIFS scenes
AudioSource	Entry point in BIFS scenes for audio streams
AudioMix	Mixes an arbitrary number of streams together
AudioSwitch	Selects and outputs an audio stream from n input streams
AudioDelay	Delays audio streams by some amount of time
AudioFx	Performs custom signal processing described in SAOL language
AudioBuffer	Clips the section of an audio stream and stores it in a buffer
Sound2D	Outputs a stream in a 3D audio scene without spatialisation
ListeningPoint	Defines position and orientation of the listener

Table 2.1: List of AudioBIFS nodes

Advanced AudioBIFS

MPEG-4 AudioBIFS version 2 [MPE01], also known as Adavanced AudioBIFS (AABIFS) provides new nodes to allow for advanced 3D audio description capabilities [VH99, VHP00]. The list of AABIFS nodes is given in table 2.2. While AABIFS is completely backward compatible with the simpler 3D audio models and nodes of AudioBIFS, AABIFS provides two new methods for improving description capabilities and realism in 3D audio scenes: the physical and perceptual approach.

The physical approach aims at describing the fine physical acoustical properties of sound sources and environments; these include: description of frequency varying directivity patterns of sound sources (*DirectiveSound* node), description of material reflectivity and transmission transfer functions (*AcousticMaterial* node) and description of reverberation times for particular areas of the 3D audio scenes (*AcousticScene* node). The physical approach of AABIFS originates from the DIVA project [HSHT96].

The perceptual approach, in contrast, performs the description of source directivity and room reverberation via orthogonal perceptual parameters. These are defined

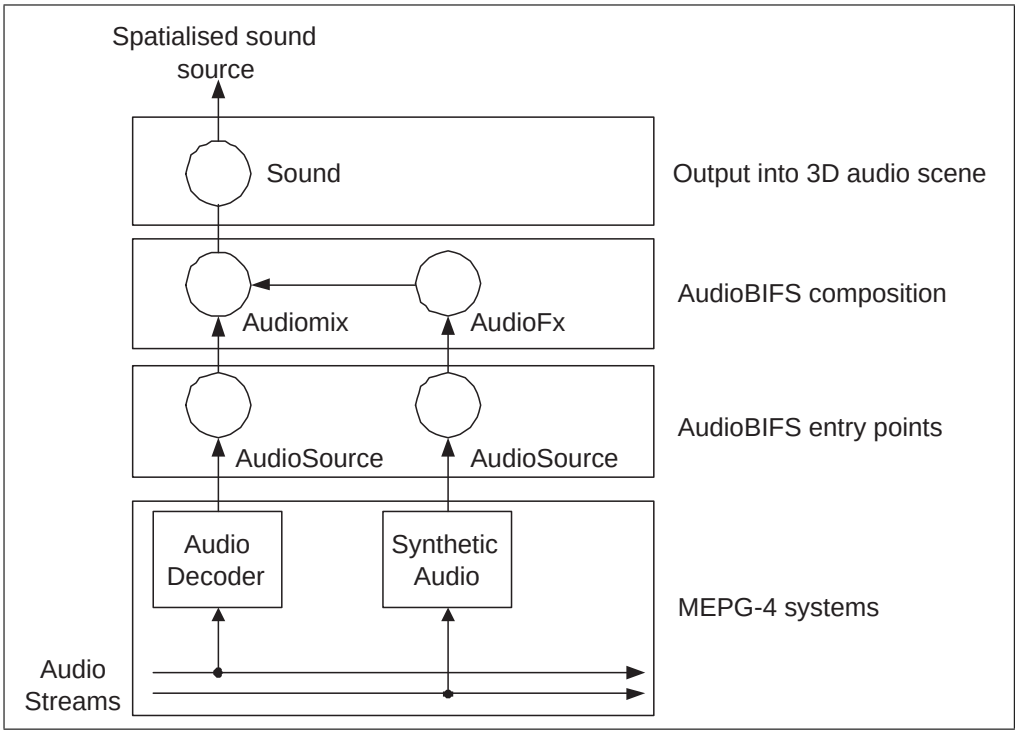


Figure 2.16: Illustration of the AudioBIFS input, composition and output nodes

in the *PerceptualParameters* node. The perceptual approach originates from Jot's reverberation models [Jot96], which are also used in IRCAM's 'Spatialisateur' [JW95].

For an AABIFS scene renderer, the computation of room reverberation from a physical scene description requires a high processing load since sound reflection algorithms (eg image model [Bor84] or raytracing [Vor89]) are computationally expensive. The perceptual approach on the other hand, requires a much smaller computation load, since reverberation can be emulated using efficient implementations such as Feedback Delay Networks (FDN) [Jot97].

Although sophisticated, MPEG-4 AABIFS has, until now, not been fully implemented in an MPEG-4 compliant decoder. Similarly, very few authoring tools exist [Vaa03]. This can be explained by the high complexity of the standard and the complex ways in which the scene nodes can interact. Several attempts have been made

[DRS+03, TJ02] to implement an MPEG-4 AABIFS capable player, however, these remain experimental implementations which use consumer electronic sound cards APIs (OpenAL [Ope] for [DRS+03] and Creative EAX [EAX] for [TJ02]). It was reported that there is great difficulty in mapping MPEG-4 AABIFS parameters to 3D audio APIs parameters [DRS+03]. The non-availability of such MPEG-4 decoder was one of the motivating factor to create a new 3D audio scene description scheme (chapter 3) which imposes less complexity on the scene renderer.

Another highlighted issue is that, in MPEG-4 AudioBIFS version 2, sound sources, despite having accurate directivity patterns and perceptual parameter descriptors, remain point sound sources (eg a flying insect or a distant sound source). Sound sources having a spatial extent (eg an insect swarm, rain, wind in trees or applause) cannot be described. Work presented in this thesis (chapter 4), with the collaboration of MPEG Audio subgroup members, resulted in the addition of sound source extent description capabilities in MPEG-4 AudioBIFS version 3, which will reach the status of Final Draft International Standard (FDIS) in 2005 (see section 4.10 for details).

Node name	**Function**
DirectiveSound	Spatialises a sound source with directivity control
AcousticMareial	Defines reflectivity and transmission properties of surfaces
AcousticScene	Defines reverberation and acoustical region delimitations
PerceptualParameters	Perceptual parameters for sources and reverberation

Table 2.2: List of Advanced AudioBIFS nodes

2.4.3 Other technologies

Java3D [Sun] is a programming language that can be used to construct dynamic 3D audio-visual environments. Java3D also follows the scene graph approach, however, being a programming language, scene animation can be performed in arbitrary ways. In terms of 3D audio capabilities, Java3D compares to VRML, in that only

point sources with ellipsoidal directivity patterns can be defined and no reverberation model exists. Being executable code, Java3D scenes must also define the methods for accessing the audio hardware. This results in poor portability and scalability in comparison to abstracted 3D audio scene description formats such as MPEG-4 AudioBIFS, VRML or XML3DAUDIO (section 3).

Another method [PL03] aimed at extending the Synchronized Multimedia Integration Language (SMIL) [SMI] to describe 3D audio scenes. SMIL is developed by the W3C consortium and is used to describe 2D multimedia content such as slide presentations etc. While it is possible to extend SMIL[14], new data types for handling 3D coordinates are required, since SMIL was designed as a 2D multimedia presentation format only. Temporal information in SMIL is also decentralised since the playing times of media are located in the fields of the audio objects. Finally, a technique for describing 3D audio scenes based on XML is proposed in [HDM03] wich is also based on a traditional scene graph approach; disadvantages of the scene graph model were highlighted in section 2.4.1 and are further detailed in chapter 3.

2.4.4 Summary of 3D audio scene encoding approaches

Channel and object oriented methods for transmitting 3D audio scenes were reviewed. While channel oriented techniques require simple terminals, they suffer from non-interactivity and poor scalability of the 3D audio scenes; this requires several encoded 3D audio scene versions for different terminal configurations (headphones, stereo or 5.1 speaker setup etc.). In contrast, object oriented techniques, by providing a higher level of abstraction and not describing the details of the rendering process, allow transmission of the same 3D audio scene to a wide range of terminals.

Problems associated with the scene graph model were then highlighted. The scene graph approach aggregates scene content, structural and temporal data; this results in complex and poorly re-usable 3D audio scene descriptions.

The VRML and MPEG-4 standards were reviewed, while VRML has only poor

[14]since it is based on XML

3D audio description capabilities, MPEG-4 AudioBIFS provides advanced features for describing complex and intricate 3D audio scenes.

2.5 Spatial auditory perception

Several spatial auditory cues are now reviewed. It is important to review these since 3D audio rendering technologies aim at artificially reproducing these spatial cues. Sound source localisation and distance perception are first reviewed, followed by the review of lesser known percepts such as the perception of sound source orientation and occlusion. The perception of sound source extent is reviewed in section 2.6.

2.5.1 Localisation

Single sound source

The Duplex theory established at the start of the 20th century by Lord Rayleigh [Ray07] states that, for a single sound source, Inter-aural Time Differences (ITD) and Inter-aural Level Differences (ILD) cues are used by the brain to perform a coarse 'left or right' localisation, this is known as 'lateralisation' [Bla97]. For sound sources not located in the median plane (where there is no inter-aural time and intensity differences), ITD is computed by the difference in the times of arrival of the sound wave reaching the two ears. ITD cues are used mainly below 1.5kHz where the phase of signals is easier to compute (Fig. 2.17a). In the case of sounds which have varying amplitudes, it was found that ITD cues can also be used at high frequencies [Bla97] as the amplitude envelope of the signal can have different times of arrival at the two ears, this is however a less dominant cue than ITD at low frequencies.

For frequencies above 1.5kHz, ILD cues are predominantly used instead since the head creates a shadowing effect; this attenuates the sound wave intensity reaching the shadowed ear (Fig. 2.17b). Shadowing effects tend to occur only above 1.5kHz since the wavelength of the sound wave becomes smaller than the size of the head. Below

1.5kHz, the sound wave is diffracted and bent around the listener's head without much attenuation (Fig. 2.17a). There is, however, no clear cut frequency at which ILDs are used instead of ITDs, but rather a transition zone ranging between 1.5 and 3kHz [Bla97, Beg94].

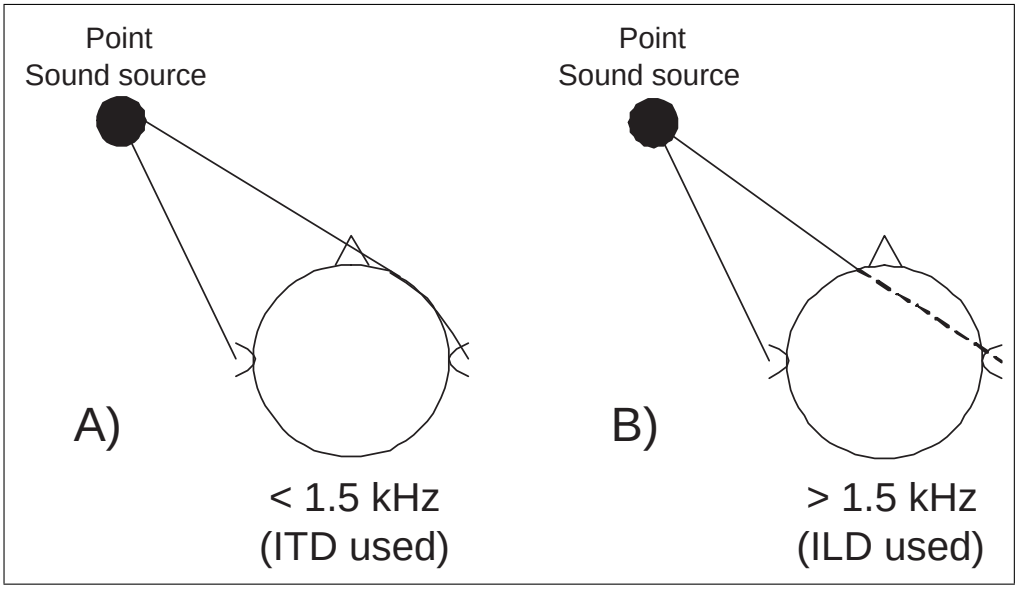


Figure 2.17: a) Use of Inter-aural time differences (ITD) at low frequencies, b) Use of Inter-aural level differences (ILD) at high frequencies

The limit of the Duplex theory, however, is that it does not explain localisation of sound sources located above, below or at the front and back of the head. Is was only after the 1960s [Bat67] that the frequency filtering effects caused by the torso, head and pinna were thought to have a contribution to the localisation of sound sources, thus completing the Duplex theory. These frequency filtering effects are dependent on the spatial position of the sound source in relation to the listener's head and are described by Head Related Transfer Functions (HRTF). HRTFs are unique to each listener and correspond to an individual 'ear print' [Beg92a].

A great amount of research has been carried out on HRTF based localisation [Car96], however, it was later demonstrated by numerous experiments [Bla97] that

unconscious head rotations also help in source localisation by removing certain ambiguities, especially front/back confusions [KJM03]. When designing a 3D audio system that uses binaural spatialisation, it is therefore required to use HRTFs that are identical or close to that of the user and a head tracking device should be used [WDO97].

The type, duration and onset time of the signal emitted by the sound source also influences spatialisation accuracy [Har83]. It is well known that constant pure sine tones are difficult to localise whereas broad band and signals with fast onsets and short durations are easier to localise [Beg94].

Multiple sound sources

In the case of multiple sound sources, the listener hears a superposition of sound sources (Fig. 2.18). Multiple sound sources occur, for instance, in stereo and multichannel speaker systems. If the signals emitted by several sound sources are coherent[15], then the 'summing localisation' phenomenon applies [The80, Bla97]. The listener then perceives a single phantom sound source which direction depends on the intensity gains of each sound source [Ger92b] (Fig. 2.18). The direction of the phantom sound source can be estimated in the following way: given a number of N sound sources placed equidistantly from a listener and producing coherent signals, the velocity vector giving the direction of the phantom sound source can be computed; this theory was first proposed by Makita [Mak62]. If G_i are the intensity gains for each speaker and $\vec{u}_i$ the directions of the speakers pointing from the centre point, the velocity vector $\vec{V}$ is the vector sum of the gains normalised by the scalar sum of the gains [Mak62]:

$$\vec{V} = \frac{\sum G_i \vec{u}_i}{\sum G_i} \tag{2.1}$$

And the scaled direction vectors are:

[15] That is, if they are statistically identical, see 2.8.2 for a mathematical definition

$$\vec{g_i} = G_i \vec{u_i} \tag{2.2}$$

The velocity vector is useful for describing summing localisation at frequencies below 700Hz [Mak62, Dan00]. However, for frequencies between 700Hz and 5kHz, it is suggested by Gerzon [Ger92e] to compute the energy vector $\vec{E}$ defined by the vector sum of the squared G_i gains normalised by the sum of the squared gains:

$$\vec{E} = \frac{\sum (G_i)^2 \vec{u_i}}{\sum (G_i)^2} \tag{2.3}$$

These two theories are complementary [Dan00] and it was suggested that for frequencies above 5kHz, frequency filtering effects caused by the pinna are used by the binaural system to further localise the phantom sound source [Beg92a]. In the case of incoherent or partially incoherent signals being emitted by multiple sound sources, summing localisation does not occur and the spatial extent (i.e. its perceived size) of the sound source is instead affected, this is reviewed in detail in section 2.8.

Localisation in reverberant conditions

In reverberant conditions, the direct sound, as well as a multitude of reflections, reach the listener (Fig. 2.19). Despite this, the binaural system is able to correctly locate the direction of the sound source amid multiple reflections (which act as phantom sound sources). Preservation of sound localisation in reverberant spaces is possible thanks to the precedence effect [Ken95, Bla97] which is also known as the 'law of the first wave front' or 'Haas effect'. The precedence effect acts as a temporal mask which inhibits the perception and localisation of delayed sound replicas within roughly 50ms of the first incoming sound, if the later are quieter than the original sound. The precedence effect is also affected by binaural differences and spectral filtering of the listener head and torso[16] and is stronger in the horizontal plane [LRYH97].

[16]The Head Related Transfer function HRTF [Beg92a]

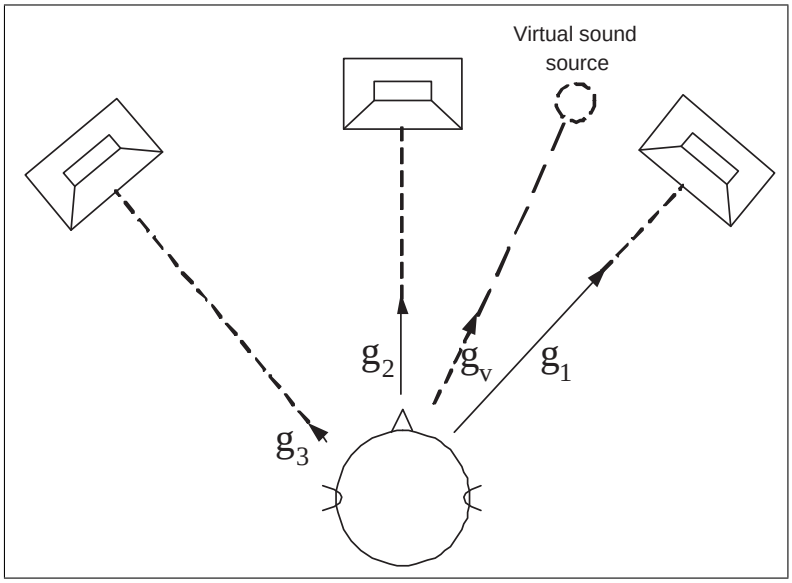


Figure 2.18: Summing localisation results in the localisation of a phantom sound source in the presence of multiple coherent sound sources

Delayed sound replicas, however, affect the perceived timbre and spectral content of the sound source due to comb filtering effects [TSA95] and also affect perceived spatial properties of the sound source; these are reviewed in 2.9. The precedence effect is further studied in the context of sound source extent perception (section 2.9.1).

2.5.2 Distance perception

In non-reverberant conditions, the most basic cue used for determining source distance is the diminution by 6 dB of sound intensity for every doubling of distance [Beg91b]. Without an absolute reference of the source intensity, however, this cue can be deceptive. Experiments on source distance perception in open fields showed that sound source distances tended to be underestimated [Nie93].

Another cue which occurs for source distances inferior to approximately two meters is the curvature of the wave front[17] which tends to boost low frequencies [Beg94] (i.e.

[17]Having a spherical propagation, the sound wave is highly curved at low distances and planar at high distances

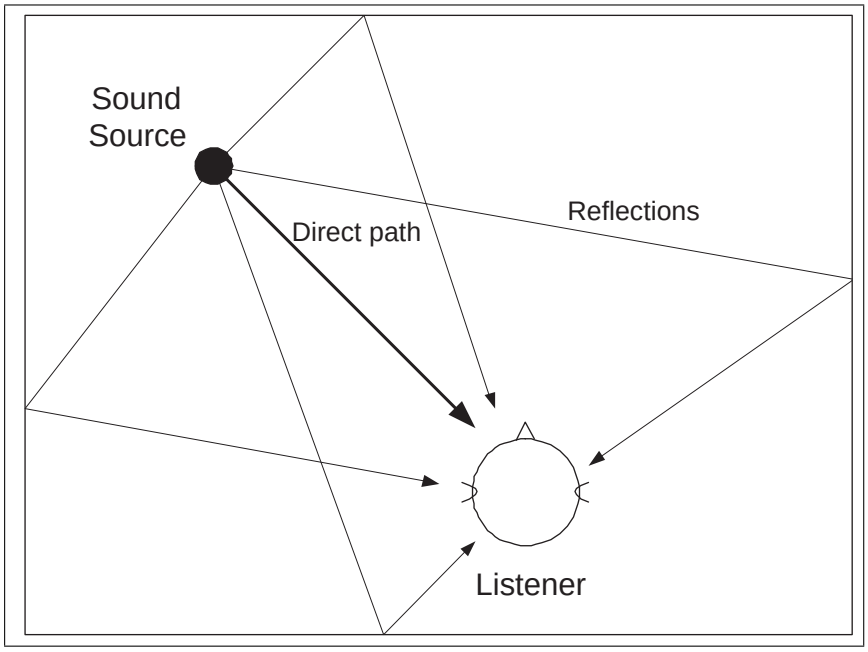


Figure 2.19: In reverberant conditions, localisation of the main sound source is preserved thanks to the precedence effect which inhibits the perception of reflections which reach the listener after the direct sound

the so called near-field effect); this cue is used only at short source distances. At distances greater than two meters, the wave front is more planar than spherical and this cue does not apply. Instead, at large source distances, the level of low-pass filtering caused by sound propagating in the air medium [Har66, BSZ95] is used as a source distance cue, albeit with not great precision [Nie93].

In reverberant conditions, the reverberant to direct sound ratio (R/D) is an accurate cue to determine source distance [Wag90, Har83, Bek62, OFR02]. Chowning [Cho71] proposed a technique by which the R/D ratio is varied to control source distance artificially. This technique is used in the 3D audio rendering system described in chapter 5. Other spatial auditory percepts are now briefly reviewed.

2.5.3 Other percepts

Some experiments studied the ability of subjects to perceive the orientation of sound sources [Neu01]. It was shown that this ability was improved when the sound source was located in front of subjects[18], for dynamic (i.e. rotating) sound sources and at short sound source distances.

Other experiments studied to perception of sound source occlusion by obstacles [FBAA03]. These experiments showed that sound source occlusion produced shifts in sound source localisation because, due to sound diffraction effects, subjects localised the edge of the obstacle instead of the occluded sound source.

Other spatial auditory percepts include the use of the Doppler effect to assess the speed of moving sound sources [Cho71] and the perception of the size and material of a room from its reverberation pattern [Kut86]. It was suggested in [Kel62] that sound reflections may also be used to locate obstacles and walls (i.e. echolocation), especially in the case of blind listeners.

2.5.4 Summary

Several spatial auditory cues were reviewed. It was shown that the binaural system and the brain use a wide range of techniques to perceive spatial sound fields. Some cues such as source localisation have been extensively studied and are now well understood, however, more research needs to be performed on other lesser studied percepts such as the perception of apparent sound source extent. A review of this percept is now given. Contributions of this thesis to the understanding of this percept are presented in chapter 4.

[18] Where sound localisation precision is maximum

2.6 Introduction to sound source extent perception

The spatial extent of a sound source is an auditory percept which can be defined as the perceived width, size or massiveness of the sound source. Sound sources such as a beach front, a waterfall and wind blowing in trees, commonly exhibit a spatially extended auditory image. In contrast, a flying insect is perceived as a small, point-like sound source. Apparent source extent is an important auditory percept which can provide information on the physical dimensions, geometry and distance of sound emitting objects.

The perceived extent of a single sound source is first studied (section 2.7), it is shown that the perceived size of a single sound source is called tonal volume and is based on diotic cues. The apparent extent of multiple sound sources (section 2.8) is then studied; it is shown that the perceived extent of multiple sound sources is based on dichotic cues and is mainly dependent on the coherence between the sound source signals and their positions. Sound source extent and other related percepts are then studied in the context of room reverberation (section 5.4.8).

2.7 Apparent size of a single sound source

The perception of the size of a single sound source is now reviewed. When studying the perceived size or massiveness of one sound source, the historic term 'tonal volume' [Ric16] has often been used. Tonal volume is a complex auditory and cognitive phenomenon and its mechanisms have, to this date, not been fully understood [Cab02, Bla97]. Békésy [Bek62] suggested that tonal volume is an auditory impression created by excitation of the basilar membrane. The author believes that the origins of tonal volume could be related to evolution theories since it can convincingly imply massive (and possibly dangerous) objects in close proximity (such as a large animal etc.). On the same lines, it was suggested that sound localisation was better in the

horizontal plane than in the vertical plane because, in the past, horizontal sound localisation was more useful for animal hunting.

It is important to note that the tonal volume of a sound source is an auditory *illusion* and does not necessarily correspond to the actual physical dimensions of the sound source; this is depicted in Fig. 2.20. Thus, tonal volume is not a spatial cue. Instead, the perception of tonal volume depends on diotic cues[19] such as: pitch (section 2.7.1), loudness (section 2.7.2), signal duration (section 2.7.3), and signal type (section 2.7.4). For example, the sound of a distant siren, despite the small apparent physical dimensions of the siren when seen from a distance, can still exhibit a massive size due to the high intensity of the emitted signal.

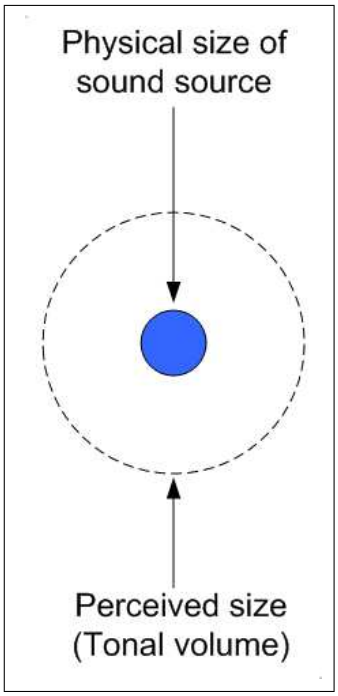


Figure 2.20: Illustration of the difference between the physical size of a sound source and its perceived tonal volume

Research on tonal volume, started at the beginning of the 20th century [Ric16, Bor26] found that the perceived size of a sound source is a function of pitch, loudness

[19]That is, cues that are simultaneously present at both ears

and signal duration of the source signal; this research has traditionally been performed on headphones with diotic stimuli. Stevens [Ste34] suggested that tonal volume was one the most important auditory attributes of sound sources, alongside pitch and loudness. Using trans-dimensional scaling, Stevens [Ste33] also demonstrated that tonal volume is an *independent* auditory attribute, although it varies with other attributes such as loudness and pitch. Guirao and Stevens [GS64] identified 'auditory density' as yet another independent attribute of sound sources. Auditory density refers to "the apparent compactness, concentration, or hardness of a sound" [GS64]. The attributes of the source signal which affect tonal volume are now reviewed.

2.7.1 Effect of pitch on tonal volume

The tonal volume of pure sine tones of different frequencies was studied, among others, by Perrot [PMS80], Stevens [Ste34, Ste33] and Boring [Bor26]. These authors found that an increase in frequency induced a smaller tonal volume. Guirao and Stevens [GS64] also found that source density or compactness increased with frequency. Perrot [PB82] asked subjects to estimate the absolute size of pure tones on an arbitrary scale; his results which are depicted in Fig. 2.21 show that tonal volume decreases with increasing stimulus frequency. In these experiments, the intensity levels of the presented stimuli were equalised for perceived loudness at the different stimulus frequencies to account for the frequency varying sensitivity of the human ear; this would have otherwise caused experimental errors, since tonal volume is affected by loudness (section 2.7.2).

2.7.2 Effect of loudness on tonal volume

Stevens [Ste34], Cabrera [Cab02] and others studied the effects of the loudness of pure sine tones on tonal volume. They found that an increase in loudness resulted in a larger tonal volume. Perrot [PB82] obtained the same results for broadband noise;

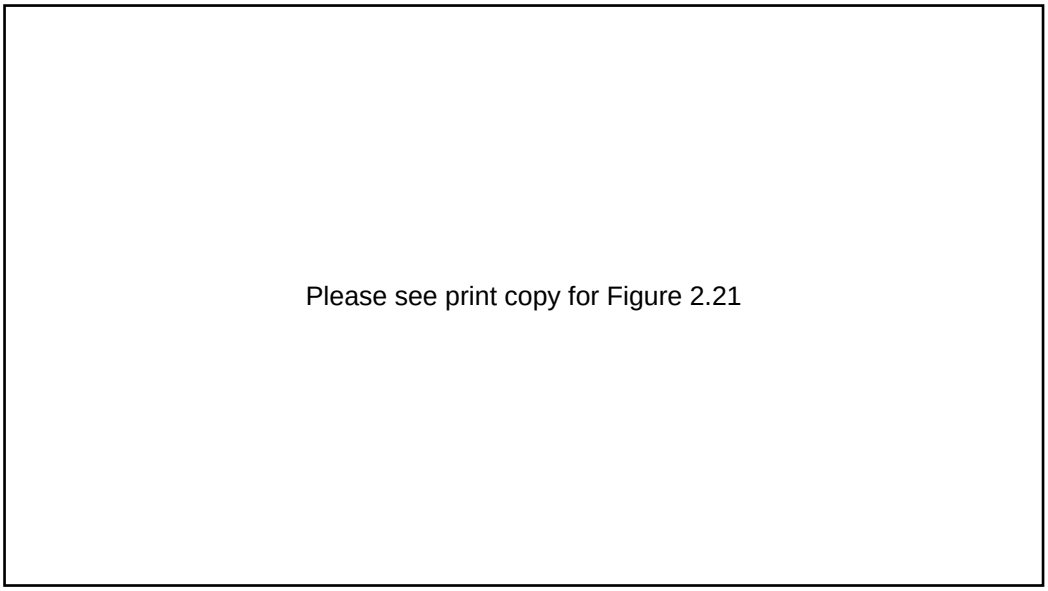


Figure 2.21: Decrease in tonal volume for an increase in frequency of a pure sine tone, at three stimulus durations (reproduced with permission from [PB82])

these are shown in Fig. 2.22. Thomas [Tho52] again noticed an increase in tonal volume for an increase in loudness of narrow-band noise stimuli. The bandwidth of the noise stimuli used by Thomas positively influenced tonal volume more so than it influenced loudness; this highlights the fact that tonal volume and loudness are distinct percepts. Stevens [Ste33] noted that subjects were able to equate the tonal volume of a loud high frequency tone with that of a soft low frequency tone. However, in a similar experiment, Thomas [Tho52] found that subjects had difficulties performing the same task between pure sine tones and broadband noise.

2.7.3 Effect of duration on tonal volume

Perrot [PB82] [PMS80] found that a longer stimuli duration positively influenced tonal volume (Fig. 2.21) and suggested that previous incoherences in tonal volume

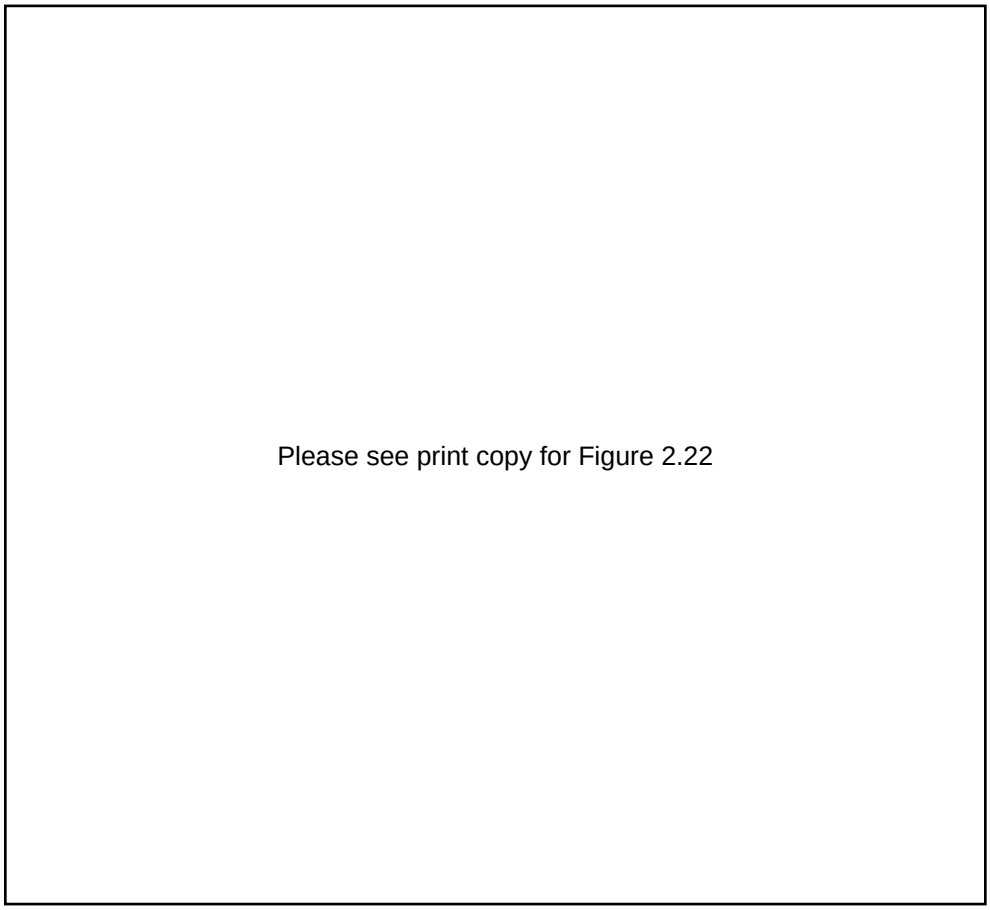


Figure 2.22: Increase in perceived tonal volume with increase in stimuli loudness and duration (reproduced with permission from [PB82])

experiments were due to a lack of control of stimuli durations. In his experiments, Perrot noticed that subjects could perceive a gradual increase in tonal volume for constant pure sine tones over periods as long as five minutes. This phenomenon, called by Perrot "the expanding-image effect" [PMS80], cannot be attributed to an increase in perceived loudness with duration since the latter occurs only in the first few hundred milliseconds [Bla97], where signal integration is performed to compute signal loudness. This finding corroborates the fact that tonal volume and loudness perception are indeed independent mechanisms.

2.7.4 Effect of signal type on tonal volume

In experiments on tonal volume, mainly pure tones and noise signals have been used. However, the timbre and type of signal emitted by the sound sources can also affect tonal volume [Bek62]. Studying the effects of signal type and timbre on tonal volume is a complex problem and a generic model is difficult to establish. For complex and natural sound sources, tonal volume may also partly depend on complex cognitive mechanisms such as memory and familiarity with the sound source. For instance, the sound of a boat horn tends to be associated with a physically large object.

2.8 Apparent extent of multiple sound sources

The perception of the apparent extent of multiple sound sources is now reviewed. Concepts highlighted in this section are essential to a technique used to render artificial sound source extent in 3D audio displays; this technique is described in section 2.12 and is subjectively tested in several experiments described in chapter 4.

Multiple sound sources can, under certain conditions that are highlighted in this section, be perceived as a single broad sound source having a particular apparent extent. A swarm of bees, for instance, can be perceived as a single broad auditory event, despite of being composed of a multitude of point sound sources (i.e. bees). Other examples of multiple sound sources that are merged into single sound sources are: wind blowing in trees, running water, rain, applause, a crowd etc. This merging effect called 'binaural fusion' by Sayers and Cherry [SC57] relates to the phenomenon that, under certain conditions, the brain perceptually merges several independent sound sources into a single sound source. Conditions for binaural fusion are reviewed in section 2.8.5.

It is important to note that the perceived *extent* of multiple sound sources is fundamentally different from the perception of the size[20] (section 2.7) of a single sound source in that its underlying auditory mechanisms are based on *dichotic* cues

[20] which was called 'tonal volume' and depends on diotic cues (eg pitch, loudness etc.)

instead of diotic cues. Besides, the perception of the extent of multiple sound sources is spatial (i.e. based on localisation) and is not a dimension-less auditory illusion like tonal volume. Dichotic cues are derived from fine dissimilarities between the signals reaching the left and right ears. Such dissimilarities can be measured with the inter-aural cross-correlation coefficient (IACC) which is defined in section 2.8.2. From experience, it is known that the IACC is highly responsible for the impression of spacious and broad auditory events (see section 2.8.4).

Another major difference between tonal volume and sound source extent is that, while tonal volume is always measured on a one-dimensional scale, the extent of multiple sound sources can exhibit one, two or three dimensions. The multi-dimensional nature of source extent is reviewed in section 2.8.6.

2.8.1 Overview of the effect

The main factor affecting the perceived extent of multiple sound sources is the level of coherence between the sound sources [Ken95]. If multiple sound sources emit coherent signals and are synchronised, summing localisation[21] occurs and consequently, only a narrow sound source is perceived at the centre of gravity of the sound sources (Fig. 2.23a). In theory, several narrow centre of gravities may be perceived at different frequencies, as it shown by equation 2.1 and 2.3 that summing localisation is affected by signal frequency. The position of the narrow sound source then depends on the positions and intensity gains of the sounds sources. This is equivalent to amplitude panning performed between several speakers [Pul99]. Alternatively, if several sound sources emit coherent signals which have different times of arrival at the listener ears[22], comb filtering is observed. This leads to sound colouration and, as parts of the perceived spectrum of the sound source are boosted or attenuated, the perceived source extent is modified[23].

[21] summing localisation was reviewed in section 2.5.1
[22] Within roughly 80ms, otherwise delays are perceived as echoes
[23] Frequency dependence of tonal volume was reviewed in 2.7.1

In contrast, if the signals emitted by the sound sources are incoherent, the binaural system is allowed to perceive them as distinct auditory streams [Ken94]. In reality, however, if the sound source signals are *cognitively* related, the brain, at a higher level, perceptually merges the sound sources into a single auditory stream [Bla02]. This, in turn, results in the perception of a spatially wide sound source (Fig. 2.23b), the extent of which, depends on the positions of the sound sources and the level of coherence between them. To be cognitively related and grouped, the sound sources may emit perceptually similar signals[24]; other conditions for grouping several sound sources into a single source are highlighted in section 2.8.5.

In addition, a low coherence between signals emitted by multiple sound sources reduces the coherence of the signals reaching the listener's ears (i.e. the inter-aural coherence). The relationship between inter sound source coherence and inter-aural coherence is formalised in section 2.8.3. Inter-aural coherence can be measured by the inter-aural cross-correlation coefficient (IACC) which is defined in section 2.8.2. Well documented publications link a low IACC coefficient with the perception of a broad and diffuse source extent [Gri97, Ken95] (section 2.8.4) and a feeling of spaciousness and envelopment in concert halls [BL86].

The effects of a low coherence between sound sources are thus two-fold: it allows the perception of multiple sound sources as single auditory streams (which may then be perceptually grouped, depending on the nature of the source signals) and secondly, it reduces the IACC coefficient which, in turn, produces an impression of spaciousness.

[24]for example, similar sounding rain drops are easily merged into a single 'rain' auditory stream

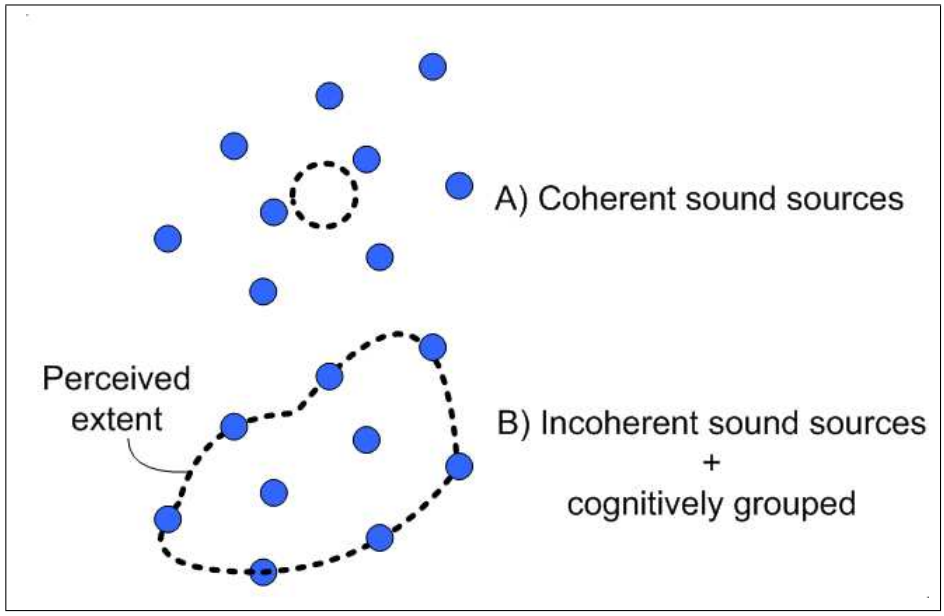


Figure 2.23: Illustration of the extent of multiple sound sources: a) Coherent sound sources result in a narrow source extent at the centre of gravity, b) Incoherent sound sources result in a broad extent

2.8.2 Definition of the inter-aural cross-correlation coefficient (IACC)

The inter-aural cross-correlation coefficient (IACC) coefficient is employed to measure the degree of similarity between signals reaching the left and right ears. In acoustical engineering, the IACC coefficient is commonly used to estimate spaciousness and listener envelopment in concert halls (see section 5.4.8). The IACC coefficient is also defined by an ISO standard [ISO97]. The IACC coefficient is defined as the maximum absolute value of the normalised cross-correlation function in turn defined as:

$$IACC(\tau) = \frac{\int_{-\infty}^{+\infty} s_L(t-\tau)s_R(\tau)dt}{\sqrt{\int_{-\infty}^{+\infty} s_L^2 dt \int_{-\infty}^{+\infty} s_R^2 dt}} \tag{2.4}$$

Where $s_L(t)$ and $s_R(t)$ are the ear canal signals at the left and right ears. The normalised cross-correlation function is bounded between -1 and 1. A cross-correlation coefficient of +1 indicates that $s_L(t)$ and $s_R(t)$ are coherent (i.e. identical) signals. A cross-correlation coefficient of -1 indicates that $s_L(t)$ and $s_R(t)$ are coherent signals with a 180 degree phase shift. A cross-correlation of 0 indicates that $s_L(t)$ and $s_R(t)$ are incoherent (i.e. dissimilar) signals. Intermediate values indicate partial coherence or incoherence between $s_L(t)$ and $s_R(t)$.

The inter sound source cross-correlation coefficient (ISCC) can also be obtained from equation 2.4 by replacing the $s_L(t)$ and $s_R(t)$ signals by the signals of two sound sources $x(t)$ and $y(t)$.

Equation 2.4 defines the full-band IACC coefficient, however it is possible to computer the IACC in different frequency bands[25] as it is known that inter-aural correlation in a reverberant room varies with frequency [PRB95, PBR95].

2.8.3 Relationship between the inter sound source correlation coefficients (ISCC) and the IACC

To study the link between the inter sound source correlation coefficients (ISCC) between multiple sound sources and the inter-aural correlation coefficient (IACC), it can be seen that, in anechoic conditions, the signals arriving at the listener's left and right ears are the sums of the source signals convolved with the Head Related Transfer functions (HRTF) for the left and right ears, respectively:

$$L(t) = \sum_{k=1}^{N} H_{L_k} * s_k(t) \tag{2.5}$$

$$R(t) = \sum_{k=1}^{N} H_{R_k} * s_k(t) \tag{2.6}$$

[25]The critical bands of the ear or third octave bands for instance

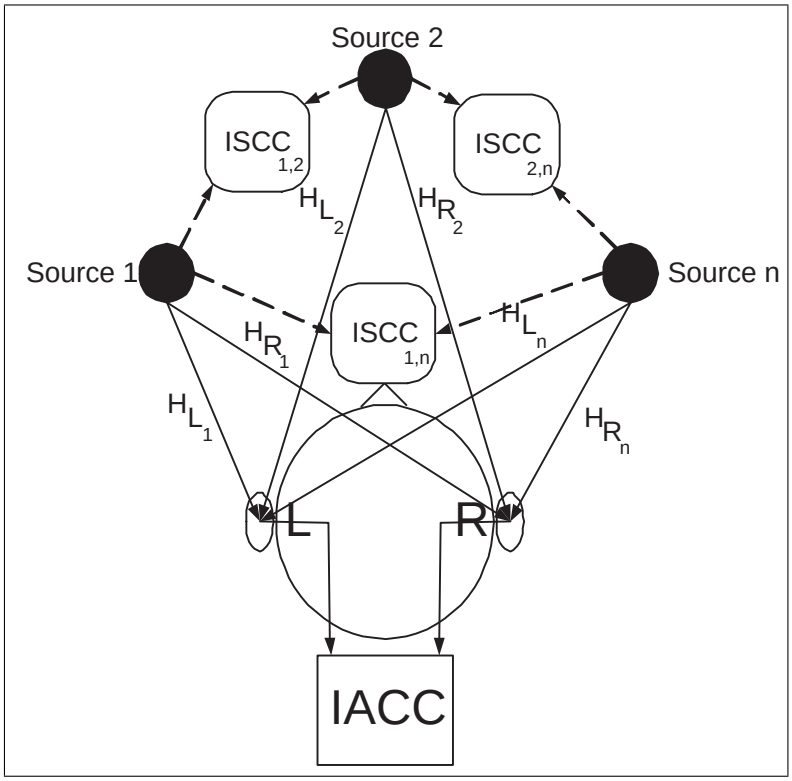


Figure 2.24: Relationship between the inter-source cross-correlation coefficients (ISCC) and the interaural cross-correlation coefficient(IACC)

Where $s_k(t)$ are the signals generated by N point sources and H_{L_k} and H_{R_k} are the HRTF functions for the left and right ears which dependent on the positions of the respective sound sources in relation to the listener; this is depicted in Fig. 2.24.

It can be seen that if the $s_k(t)$ signals are highly correlated or identical, the IACC coefficient only depends on the decorrelation caused by the HRTF functions; this decorrelation being weak. For coherent signals emitted by the sound sources, the resulting IACC coefficient is thus high. On the other hand, if the $s_k(t)$ signals are totally incoherent (ISCCs=0), the IACC value decreases, but does not reach zero due to coherence re-introduced by the HRTF functions. It should be noted that in echoic conditions, room reverberation tends to further reduce the IACC coefficient [TSA95].

Blauert, on the topic of the relationship between IACC and ISCC, states “As a rule, the range of variation of the degree of coherence of the ear input signals is

smaller than that of the signals at the sound sources" [Bla97] p240.

Kurozumi and Ohgushi [KO83] carried out an experiment where they measured the IACC coefficient at a dummy head microphone in function of the correlation coefficient between two noise sequences played on two speakers in an anechoic chamber. They measured IACC coefficients of 0.93, 0.02 and -0.88 for ISCC coefficients of 1, 0 and -1, respectively. This is conform to Blauert's statement and equations 2.5 and 2.6.

It can be thus be said that the inter sound source correlation coefficients (ISCCs) directly affect the inter-aural cross-correlation coefficient (IACC) but that the range of variations of the IACC is always less than that of the ISCC, due to the listener's HRTFs.

2.8.4 Effects of inter sound source coherence

Headphones presentation

Chernyak and Dubrosvy [CD68] studied the effects of the cross-correlation coefficient of two broadband noise signals presented to the left and right ears on headphones. Being presented on headphones, the cross-correlation coefficient between the noise sequences directly controlled the inter-aural cross-correlation coefficient (IACC). Chernyak and Dubrosvy found that coherent noise signals (IACC=1) produced a narrow sound extent at the centre of the head. In contrast, incoherent signals (IACC=0) resulted in the 'externalisation' of two distinct sound images around the ears (i.e. binaural fusion was lost). Decreasing the IACC from 1 to 0.4 resulted in the broadening of the spatial image perceived within the head, and decreasing the IACC below 0.4 corresponded to the threshold at which binaural fusion was lost, and the presence of two distinct auditory streams emerged. The results of Chernyak and Dubrosvy are summarised in Fig. 2.25.

Blauert and Linderman [BL85] carried a similar experiment with pink noise and found similar results to that of Chernyak and Dubrosvy. However, for low values

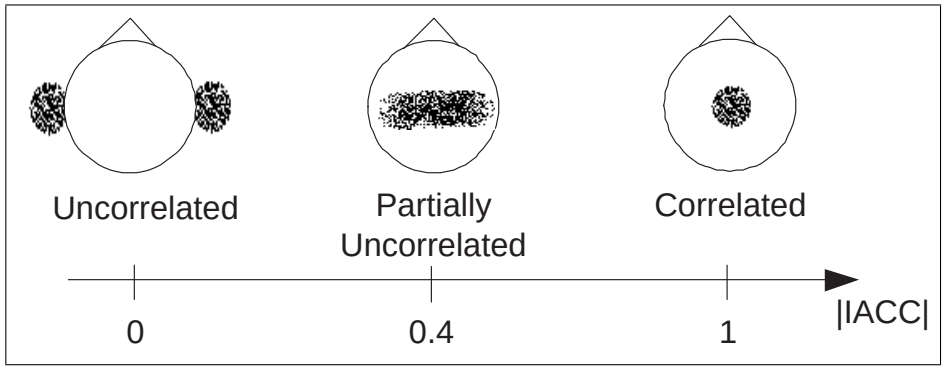


Figure 2.25: Effect of the inter-aural cross-correlation coefficient (IACC) on the apparent image width of white noise presented on headphones

of the IACC coefficient, they noticed that subjects could sometimes perceive three different auditory events: two low frequency components at the listener's ears and a narrow high frequency component at the centre of the head. This finding hints that the IACC coefficient is a function of frequency.

Two speakers

Other authors [Ken95, KO83] studied the apparent extent of white noise presented on two speakers. They found that the level of correlation between the two presented channels had a dramatic impact on the perception of sound source extent. A high correlation coefficient between two broadband noise sequences presented on two speakers resulted in a narrower central source extent. On the other hand, a low correlation value between the noise sequences resulted in an extended sound source filling the space between speakers; this effect is depicted in Fig. 2.26. Kurozumi and Ohgushi [KO83], however, never mentioned the loss of binaural fusion in their experiments, possibly because the speakers they used were spatially close enough to permit binaural fusion (see section 2.8.5). Kurozumi and Ohgushi [KO83] mentioned that negative correlation coefficient values between the noise sequences affected the distance and elevation of the apparent broad sound source. There is currently no psychoacoustic explanation of this phenomenon [Mas02] and the author was not able to reproduce

this effect in informal experiments.

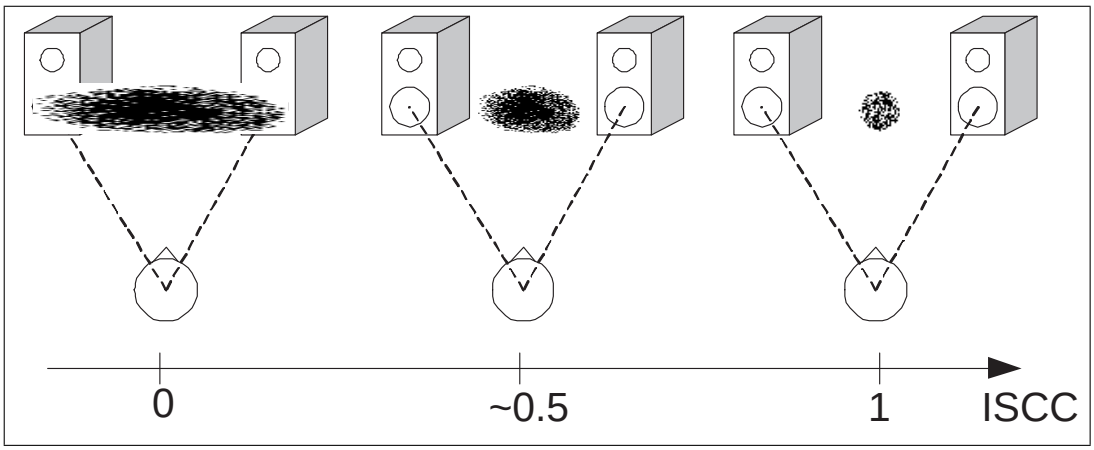


Figure 2.26: Effect of inter-channel correlation on perceived spatial extent

More than two sound sources

Damaske [Dam68] carried out an experiment where he used four speakers placed around subjects. He used band-limited noise as stimulus and varied the inter-speaker cross-correlation coefficient from 0.98 to 0.15. Again, lowering of the inter-speaker correlation coefficients reduced the IACC coefficient and subjects reported a larger and more diffuse source extent. The results of Damaske are thus coherent with the previous mentioned experiments.

In chapter 4 novel experiments are described where the extent of multiple incoherent sound sources is studied. In particular, it is studied whether listeners are able to measure an absolute source extent rather than a vague impression of spaciousness such as in the experiments described in this section.

2.8.5 Conditions for binaural fusion

Binaural fusion is a phenomenon by which several sound sources are perceptually merged into a single sound source. Binaural fusion is thus a requirement to perceive

the extent of multiple sound sources. Binaural fusion, has two levels: a (low) binaural system level which is affected by the coherence and the distance between sound sources, and a (high) cognitive level which is affected by cognitive proximity between sound sources and by more complex factors such as familiarity and attention focusing. The factors affecting binaural fusion are now reviewed.

Inter sound source coherence

Jeffress [Jef47] and Sayers and Cherry [SC57] were among the first to propose the inter-aural cross-correlation coefficient (IACC) as the main psychoacoustic factor affecting binaural fusion. Sayers proposed a model of the binaural system which performs a running cross-correlation task, which is used to fuse or separate auditory streams. He found that correlated sound sources tended to be perceptually merged; this is similar to the summing localisation effect which was reviewed in section 2.5.1.

Licklider [Lic48] performed several headphone experiments where he measured the intelligibility of speech masked by broadband noise. He noticed that when correlated speech and uncorrelated noise were used, the speech signal was fused at the centre of the head while the noise was externalised to the left and right ears; this resulted in better intelligibility than if both the speech and noise signals were both uncorrelated or correlated simultaneously.

As a rule, coherent sound sources that are spatially close are perceptually merged by the binaural system and incoherent sound sources are separated in different auditory streams (which may then be merged again at a higher cognitive level, see Fig. 2.28).

Distance between sound sources

The position of sound sources, when these emit incoherent signals, also affect the extent of the global sound source and binaural fusion. Indeed, large binaural differences such as ITD and ILD (see section 2.5.1) caused by sound sources which have a wide spatial separation tend to separate auditory streams. Effects of source separation are

summarised in Fig. 2.27. In Fig. 2.27a, two incoherent sound sources have a small angular separation and are perceived as a single auditory event; binaural fusion occurs. In Fig. 2.27b, binaural fusion still occurs but global extent is broader due to a wider separation of the sound sources. Lastly, in Fig. 2.27c, the two sound sources are too far apart and are perceived as distinct auditory events. The author also noticed the loss of binaural fusion effect due to source separation in the experiment described in section 4.3; binaural fusion was lost when two uncorrelated sound sources (emitting noise signals) were far apart more than 30 degrees on the horizontal plane.

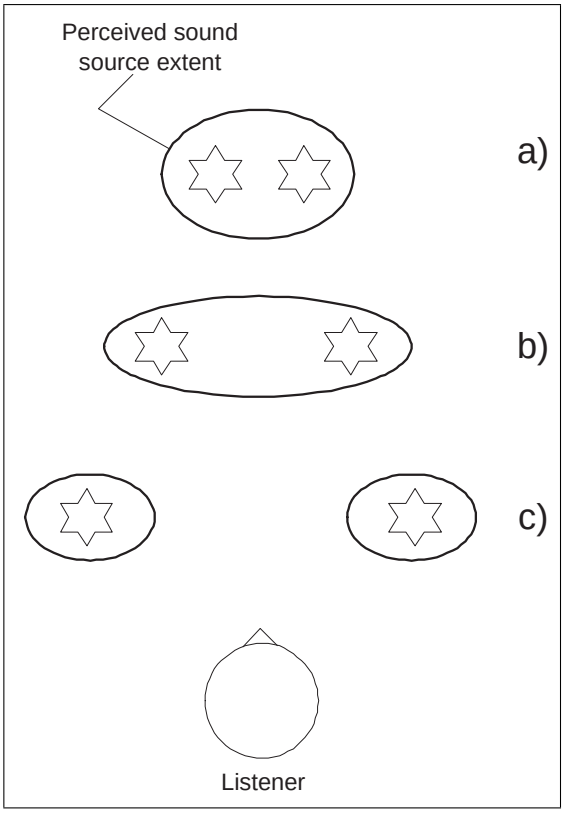


Figure 2.27: Effects of angular separation between two uncorrelated sound sources on apparent source extent and binaural fusion: a) Perception of a single narrow auditory event, b) Perception of a single broad auditory event, c) Perception of two distinct auditory events

Cognitive proximity

Finally, complex cognitive mechanisms may also influence binaural fusion. This is reflected in what is commonly known as the 'cocktail party effect' [Aro00] which was a term first introduced by Cherry [Che53]. The cocktail party effect refers to the ability of humans to consciously focus and listen to a particular conversation among many simultaneous conversations. Thus, by a conscious process, the brain is able to merge several auditory streams into a single 'background noise' auditory stream; this then helps focusing on a particular auditory stream.

Another cognitive effect which creates binaural fusion is when several sound sources, although uncorrelated, are perceptually similar and thus are merged into a single auditory stream. This is the case, for instance, of uncorrelated white noise sequences merged into a common noise source, rain drops merged into rain, persons clapping merged into applause etc. This effect is the basis of the technique for rendering sound source extent in 3D audio displays described in section 2.12 which uses decorrelation to create several replicas of a signal; these are then allowed to merge cognitively to form a single broad sound source.

A last cognitive effect which creates binaural fusion is when several sound sources are part of a larger complex object. For example, a truck emits a multitude of sounds from different locations (eg engine, exhaust, vibrating panels etc.) and can be perceived as a single broad sound object. By a conscious process, however, it is possible to focus only on one particular sound source of the complex object. This phenomenon is linked to perceptual grouping and is studied in the science of auditory scene analysis [Bre94].

Summary of the conditions affecting binaural fusion

The general model of the conditions affecting binaural fusion and perceived source extent are summarised in Fig. 2.28. In this model, four scenarios are identified:

- Fig. 2.28a: Coherent sound sources are merged into a narrow sound source

(summing localisation).

- Fig. 2.28b: Incoherent sound sources which are not cognitively related are perceived as distinct sound sources.
- Fig. 2.28c: Incoherent sound sources which are cognitively related but too far apart are not binaurally fused.
- Fig. 2.28d: Incoherent sound sources which are cognitively related and spatially close are perceived as a single broad sound source. This last scenario is used to create broad sound sources artificially in 3D audio displays (section 2.12).

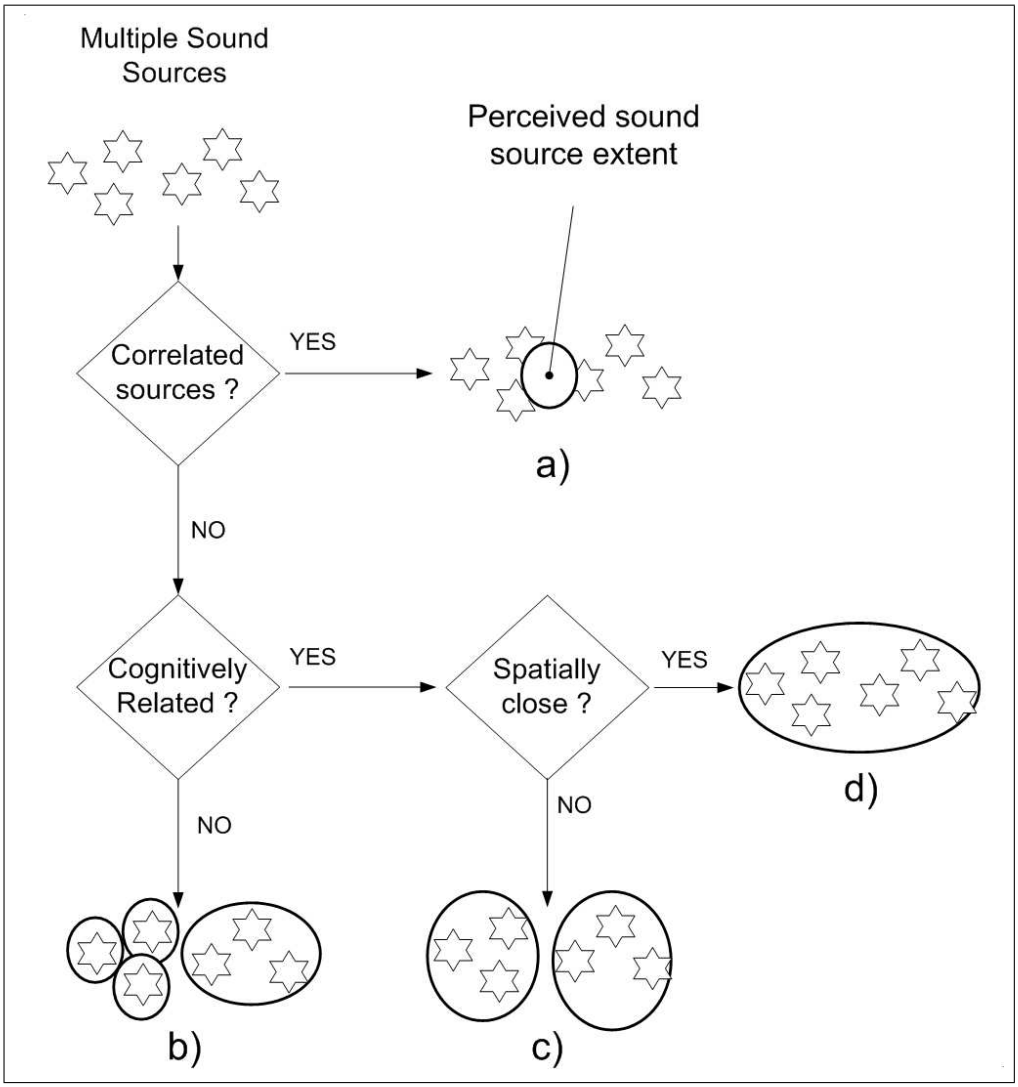


Figure 2.28: General model of the conditions affecting binaural fusion and apparent extent of multiple sound sources

2.8.6 Multi-dimensionality of sound source extent

So far, the extent of multiple sound sources has only been considered as a one-dimensional auditory percept of sound sources. It seems, however, possible that some natural sound sources can exhibit a multi-dimensional extent and that sound sources may have zero (point source), one (line source), two (surface source) and three (volume source) dimensions; some examples of such sound sources are given in Fig. 2.29.

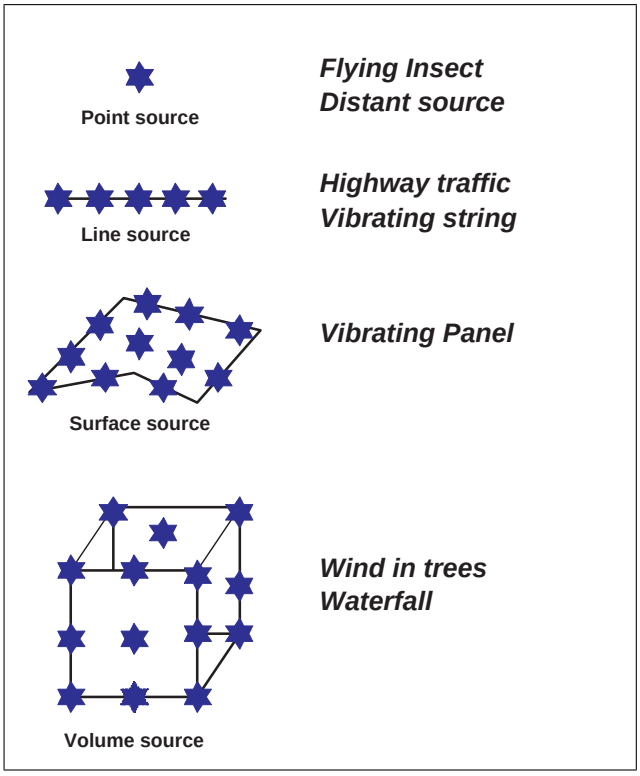


Figure 2.29: Example of one-dimensional and multidimensional sound sources

To the knowledge of the author, limited literature studied the perception of multi-dimensional sound source extent. Perrot [PB82] wrote "Auditory images may have one, two, or even three dimensions, but there is insufficient evidence to date to make such a determination". One of the first experiments going towards the study of multi-dimensional sound source extent were carried out by Perrot [PMS80]. Perrot

studied the perception of horizontal and vertical sound source extent, and found that results were more variable in the vertical axis; this could be related to a lesser sound localisation ability on the vertical axis than on the horizontal axis [Bla97]. In section 4.4, an experiment is presented where the perception of vertical and horizontal source extent is also studied.

Ruff and Perret [RP76] performed several experiments where they presented particular auditory patterns on a 10 by 10 speaker matrix. They presented several symbols, letters and numbers on their 'audio raster'. They used pure sine tones as stimuli, which unfortunately, are difficult to localise [Bla97]. By switching on and off certain speakers in sequence, particular patterns at a certain speed could be drawn on the speaker matrix. They found that subjects could identify particular sound patterns with a probability higher than statistical chance. However, it should be noted that the technique they used to create the sound patterns relied on the spatial trajectory of a single sound source rather than on a particular sound source extent. Therefore, it can be argued that their experiment was only testing the ability of subjects to perceive and memorise sound source trajectories.

Lakatos [Lak93] carried out similar experiments to that of Ruff and Perret [RP76] but using complex tones as stimuli. Lakatos found that the higher the bandwidth of the stimuli the better was the percentage of correct pattern identification. Again, since Lakatos used a single sound source drawing a pattern, it can be seen that an increased bandwidth would have increased localisation ability by subjects; thus resulting in a higher percentage of pattern identification.

Hollander [Hol94] was the first to study the perception of the spatial extent and shape of simultaneous sound sources. Hollander, however, attempted to construct auditory shapes with several coherent sound sources that were presented binaurally on headphones. As explained in section 2.8.1, coherent sound sources produce the effect of summing localisation, inhibiting the ability of the binaural system to perceive extended sound sources. For these reasons, Hollander's subjects were not able to perceive sound source shapes and his results were inconclusive.

Evreinov [Evr01] proposed a method for representing geometrical shapes using sound. Evrenivov used different frequencies to represent a point on a 2-dimensional plane. The effectiveness of this approach has not been subjectively tested, however.

Rocchesso [Roc01] studied the ability of subjects to perceive the shapes of resonators; this topic was studied previously by Kac [Kac66] and later by Kunkler-Peck [KPT00]. Rocchesso wondered whether subjects could hear the shape of a drum from the spectral content of the generated sound. Tucker [TB03] studied the perception of size, shape and material of struck plates. Although the study of the perception of resonator shape differs from multidimensional sound source extent discussed here, it is interesting to review this literature.

In section 4.5 and 4.6 novel experiments are presented which study the perception of multidimensional sound source extent forming apparent shapes.

2.9 Perception of source extent and spaciousness in reverberant environments

So far, source extent has only been only studied in anechoic conditions. In reverberant conditions, not only the direct sound reaches the listener but also a multitude of reflections having a spread of arrival times. The simplified impulse response of a room is depicted in Fig. 2.30. Room impulse response is commonly separated in the direct path signal, early reflections (within 100ms of the original sound) and late reverberation. Early reflections and late reverberation combine into psychoacoustics effects that affect sound source extent and other percepts; these percepts are now outlined.

2.9.1 The precedence effect

An important psychoacoustic phenomenon in reverberant spaces is the so-called precedence effect also known as the 'law of the first wavefront' [Mor02, Ken95, Bla97]. The

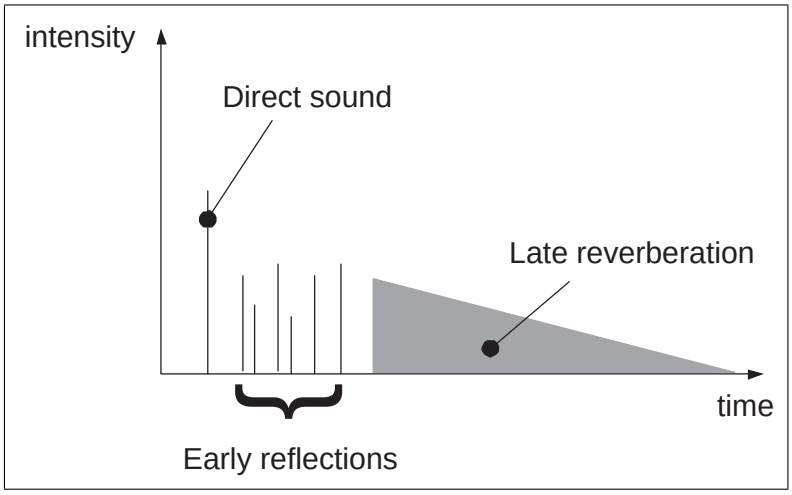


Figure 2.30: Simplified model of room reverberation

precedence effect is a process taking place in the binaural system which perceptually masks the delayed reflected replica of the original sound in a 80ms window. This is useful to preserve localisation of the original sound source amid many reflections[26]. After 80ms, a repeated sound is perceived as an echo of the original sound; this is depicted in Fig. 2.31. Although masked, early reflections, especially lateral ones, dramatically improve the impression of spaciousness [Gri96]. This relies, however, on the additional presence of late reverberation [Gri97], otherwise, small rooms would be perceived as spacious. This finding is extensively used in the design of concert halls [Ber96], so that the shape of a concert hall is chosen so as to generate as much lateral reflections as possible.

2.9.2 Spatial Impression

In the study of concert halls, a term know as 'Spatial Impression' (SI) has been introduced by Barron [Bar99]. Spatial impression is a high level and multi-dimensional auditory attribute attached to the perception of a reverberant space. Lehnert [Leh93] defines SI as "The concept of the type and size of an actual or simulated space to

[26]this was reviewed in section 2.5.1

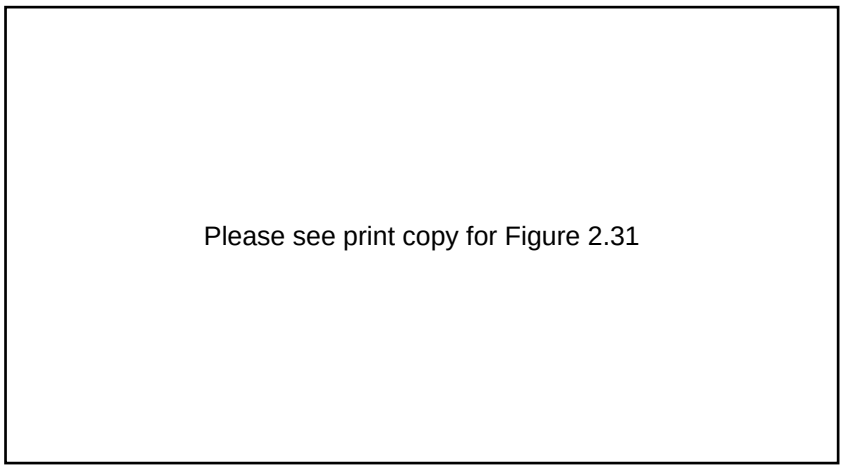


Figure 2.31: Illustration of the precedence effect (after Kendal [Ken95]

which a listener arrives spontaneously when he/she is exposed to an appropriate sound field". Spatial Impression contains: Spaciousness, Apparent source width, Envelopment and Reverberance. Fig. 2.32 describes the relationships between these four percepts. These percepts are not detailed.

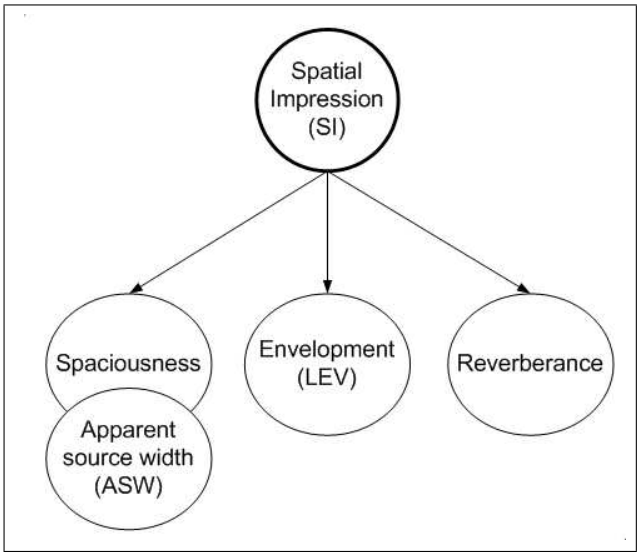


Figure 2.32: Relationship between 'Spatial Impression' and other auditory percepts

2.9.3 Spaciousness

Spaciousness is defined by "The apparent enlarged extension of the auditory event, in particular the apparent enlarged extensions of the auditory image compared to that of the visual image" [Leh93]. Griesinger [Gri97] is however cautious not to consider spaciousness and apparent source width (ASW) as the same auditory percept. Griesinger states that "In the English language a concert hall can be spacious, the reverberation of an oboe can be spacious, but the sonic image of an oboe cannot be spacious". The author partly agrees to that statement in that an oboe cannot be spacious but may still have a spatial extent.

The main factor affecting the impression of spaciousness in concert halls is the level of coherence between signals arriving at the listener's right and left ears [Ber96] which can be measured by the IACC coefficient (previously defined in section 2.8.2). In reverberant spaces, the IACC is reduced by sound reflections and diffusion which tend to produce fine dissimilarities between the signals reaching the left and right ears. An IACC value close to zero will introduce a sense of spaciousness and of a spatially large sound source; in contrast, an IACC absolute value close to 1 will produce a narrow sound image. In concert halls, a low IACC value improves the feeling of spaciousness and source width. Mason [Mas02] suggested that temporal fluctuations of the IACC coefficient could also influence spaciousness.

2.9.4 Apparent source width

In concert halls and reverberant conditions, when the distance between listener and sound source augments, the power ratio between direct sound and reverberation decreases [Cho71]. This, in turn, decreases the IACC coefficient and increases apparent source width (ASW) and localisation blur [Bla97] (Fig. 2.33). It should be noted that ASW and the apparent extent of multiple sound sources (described in section 2.8) are different percepts since ASW is due to a decrease of IACC from reverberation, while apparent source extent is based on the localisation of the multiple sound sources and

on a low level of coherence between them (which also reduces IACC) and thus, the apparent extent of multiple sound sources does not need reverberation to exist.

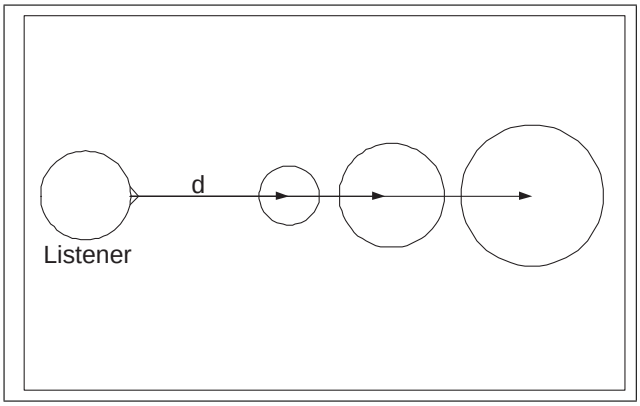


Figure 2.33: Apparent Source Width (ASW) and localisation blur increase with distance in reverberant conditions

2.9.5 Listener envelopment

Morimoto defines Listener envelopment (LEV) as "the degree of fullness of sound images around the listener" [Mor02]. Morimoto also demonstrated that listeners could perceive envelopment and apparent source width as independent auditory percepts [MM89]. In concert halls, Beranek states that "the reverberant sound that reaches the listener after 80ms is most pleasant if the listener hears it coming from all directions" [Ber96]. Listener envelopment can thus be considered as a pleasing quality of sound fields and defines the degree at which reverberation is spread around the listener.

2.9.6 Reverberance

Lastly, reverberance defines the amount of reverberation itself without referring to its spatial qualities. Lehnert [Leh93] defines reverberance as "the sensation that in addition to the direct sound, reflections are present which are not perceived as repetitions of the original signal".

2.10 Summary of sound source extent perception

The perception of sound source extent was reviewed in the context of a single sound source and multiple sound sources. It was shown that the perception of the size (i.e. tonal volume) of a single sound source is controlled by the pitch, loudness, duration and type of the signal emitted by the sound source. On the other hand, the perceived extent of multiple sound sources (if binaurally fused) depends on the position of the sound sources and the level of coherence between them. It was shown that a low coherence between sound sources, in turn, reduces inter aural coherence which is responsible for the impression of spaciousness. The perception of multidimensional source extent was then reviewed, it was suggested that certain natural sound sources may exhibit multidimensional extents. Sound source extent in the context of reverberant spaces was then reviewed, it was shown that reverberation tends to decrease the IACC, resulting in wider apparent source width and an increased impression of spaciousness.

This thesis presents in chapter 4 several contributions to the study of the apparent extent of multiple sound sources.

2.11 Sound source extent rendering techniques

Having reviewed psychoacoustic phenomena involved in the perception of sound source extent (section 2.6), this section gives a review of several techniques that can be used to create and control sound source extent artificially. Sections 2.11.1 and 2.11.2 first review precursor techniques that have been used to control sound image width in stereo recordings. The following sections (2.11.3, 2.11.4, 2.11.5) review techniques that are used to control sound source extent in the context of the Ambisonics and VBAP spatialisation techniques. A more advanced source extent rendering technique which uses decorrelated sound sources is reviewed in section 2.12.

2.11.1 Stereo sound recording techniques

Since the invention of stereophony by Blumlein in 1933 [Blu33], it has been possible to devise broad and spacious audio recordings using only two audio channels. Indeed, it was shown in section 2.8 that decorrelation between two speaker or headphone channels produced impressions of broad and spacious sound images. Audio engineers have been particularly attentive to this fact so as to produce pleasing and spacious stereo recordings.

A well known technique used to control the image width of stereo recordings is the Mid-Side (MS) technique [Rum01]. This technique uses a stereo microphone composed of a cardioid microphone facing the scene (recoding the M signal) and a figure of eight microphone perpendicular to the scene (recoding the S signal). Using a matrixing process, the left and right speaker signals are obtained:

$$L = a * M + b * S \tag{2.7}$$

$$R = a * M - b * S \tag{2.8}$$

By varying the gain a of the M signal and the gain b of the S signal, control of sound image width is achieved. Indeed, varying these gains in turn affects the inter-channel correlation since maximally correlated (a=1, b=0) or maximally decorrelated (a=0, b=1) speaker signals can be obtained. This simple technique for controlling stereo image width is also convenient since it permits to alter image width *after* a recording was made. The MS stereophonic recording technique is summarised in Fig. 2.34.

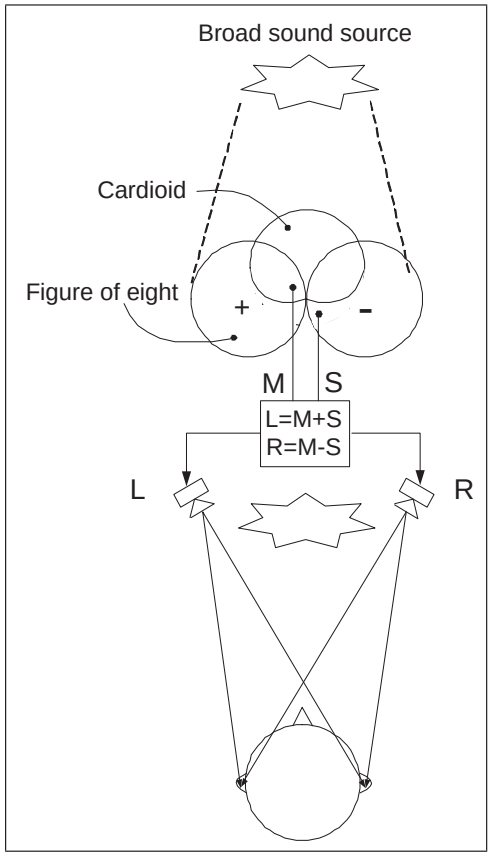


Figure 2.34: Illustration of the MS stereophonic microphone recording to capture and control image width

2.11.2 Pseudo-stereo processors

Since the 60's, several techniques were invented to produce pseudo-stereo recordings from monaural sound recordings (see [Orb70] for a review). These techniques can be used to create wide stereophonic images from monaural sound recordings and are based on altering the phase of an input monaural signal in several frequency bands differently to obtain decorrelated left and right speaker signals [Sch58, Bau63, WMK93, Yok85]; this in turn, permits a wide stereophonic image. These techniques can be considered as the ancestors of the technique described in section 2.12 which is used to render sound source extent in 3D audio displays.

2.11.3 Ambisonics W Channel boosting

A method for increasing sound image broadness in the context of Ambisonics spatialisation (see section 5.4.1) is to boost the W Ambisonics channel[27] so as to increase the amount of W signal into each speaker [Mal95]. This technique, however, is empirical and blurs the spatialised sound source so that it is more difficult to localise. Indeed, it can be seen that increasing the amount of W signal into each speaker increases the inter-correlation between speakers which, in turn, increases the inter-aural cross-correlation coefficient (IACC)[28]; this goes against the perception of a spacious sound image (section 2.8). Therefore, this technique only increases sound source blur rather than actual sound source extent.

An improvement of this technique known as W-panning was proposed by Menzies [Men02] in which he proposes to vary the gain of the W channel in function of sound source distance. Benefits of this technique over simple W channel boosting, however, were not demonstrated by perceptual evaluation.

2.11.4 Ambisonics O-Format

Another technique for rendering sound source extent (and directivity) in the context of Ambisonics spatialisation[29] is known as 'O-format' [Men02, Mal99a, Gir96]. This technique consists of encoding the extent and directivity pattern of a sound source into spherical harmonics[30] impulse responses. This technique can be considered as the inverse of the Ambisonic technique. Unlike Ambisonics, which describes the *inwards* sound field arriving at one point[31] (Fig. 2.35a), the O-format technique describes the *outwards* radiation pattern and extent of a sound source (Fig. 2.35b). Menzies [Men02] states that this consists of 'turning the signal inside out'.

To obtain the O-format information about a sound source, it was suggested

[27] which corresponds to the 0th order spherical harmonics or the omnidirectional component
[28] This effect is explained in 2.8.3
[29] see section 5.4.1
[30] spherical harmonics are the basis of Ambisonics spatialisation, see [Dan00]
[31] At the sweet spot, where the listener should be located

[Mal99a] to use a microphone array surrounding the sound source; this is depicted in Fig. 2.36. The sampled radiation pattern[32] then needs to be mathematically processed to obtain the O-format impulse responses [Men02, Mal99a]. Once described in this format, the measured source radiation pattern can be used in an Ambisonics scene by convolving the source signal with the obtained O-format impulse responses [Mal99a]. Although innovative, no perceptual evaluation of this technique has been published and the amount of literature on O-format is small.

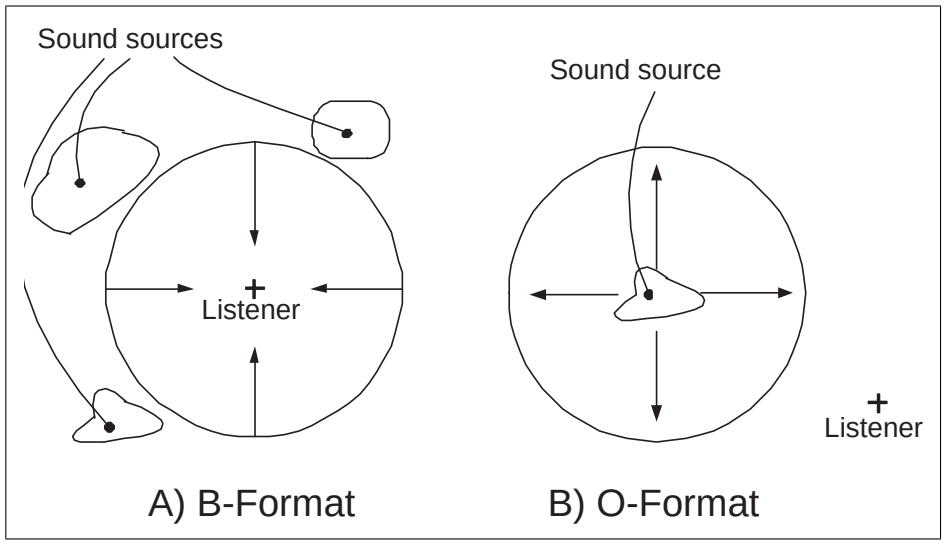


Figure 2.35: Inwards and outwards equivalence between B-format and O-format

[32]Which contains both source extent and directivity information

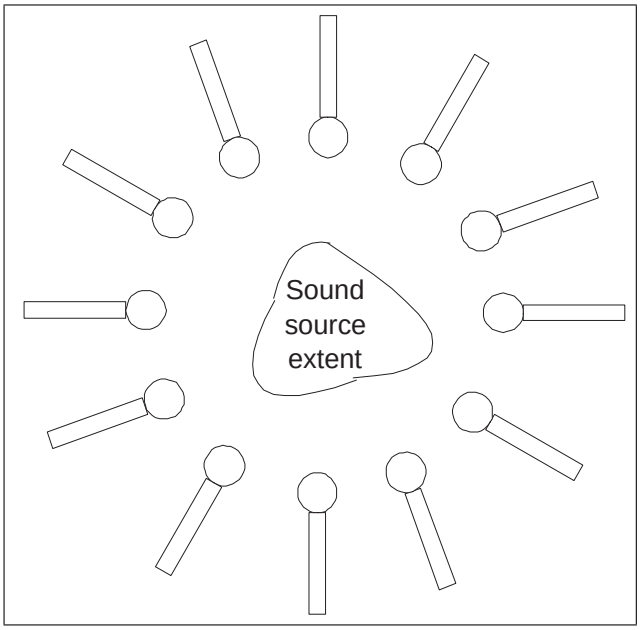


Figure 2.36: Sampling of the directivity pattern and shape extent of a sound source with a microphone array prior to O-format conversion

2.11.5 VBAP spread

Vector Based Amplitude Panning (VBAP) [Pul97, Pul01] is a spatialisation technique which is briefly reviewed in section 5.4.1. For the VBAP spatialisation technique, Pulkki [Pul99] proposed a technique to control sound source extent. To do so, he proposed to spatialise a few identical point sources around the main sound source. However, it was shown in section 2.8 that the use of identical (i.e. correlated) point sources does not affect source extent, but instead affects localisation blur. The technique proposed by Pulkki is somehow similar to the W channel boosting technique (section 2.11.3) in that it does not reduce the inter-aural cross-correlation coefficient (IACC)[33] and thus it is ineffective to render sound source extent.

[33]defined in section 2.8.2

2.12 Rendering of sound source extent using decorrelated point sources

2.12.1 Preliminary observations on natural sound sources

A technique used to control the extent of sound sources is described in [Ken95, KWM93, Sib02]. This technique relies on the observation that a physically broad sound source can be decomposed into several discrete sound sources. For instance, a vibrating panel emits sounds differently on its surfaces (depending on the vibration modes, surface material etc.). In 3D audio rendering, this vibrating panel can be approximated by a finite number of point sound sources that emit non-identical signals; this is depicted in Fig. 2.37.

Other natural sound sources, such as wind blowing in trees, a swarm of insects, a beach front, an highway are indeed composed of independent, discrete sound sources. The spatial distribution of these sources then defines the overall extent and geometry of the global perceived auditory event. For instance, beach front and highway sound sources, if facing them, appear as line sound sources, and wind blowing in trees or a swarm of insects appear as 2D or 3D extended sound sources.

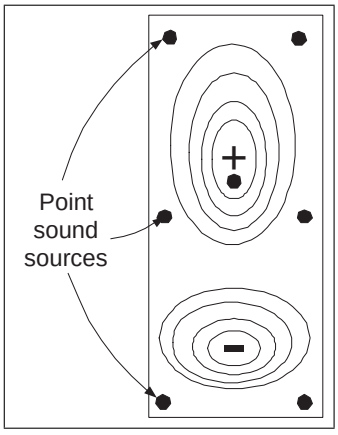


Figure 2.37: Decomposition of a vibrating panel source into several point sound source

2.12.2 General principle

From the observations made on natural sound sources (section 2.12.1), spatially large sound sources can be rendered in 3D auditory displays by spatialising[34] several discrete sound sources at different positions and which emit *decorrelated* signals [PB03, PS03, Ken95, Sib02]. Indeed, for psychoacoustic reasons that were highlighted in section 2.8, if correlation is high between the spatialised point sources, the binaural system perceives them as a single, narrow auditory event.

In contrast, if the signals emitted by the point sources are uncorrelated, the binaural system perceives the point sources as distinct sound sources. In reality however, if the point sources are cognitively related (i.e. same type of signal) and spatially close, at a higher cognitive level, the decorrelated point sources are merged in a single, spatially large, sound source; this merging effect called binaural fusion was reviewed in section 2.8.5.

2.12.3 1, 2 or 3D source extent

By controlling the position of the discrete point sources, it was suggested [Sib02] that the present technique could be used to devise 1D, 2D and 3D extended sound sources. In reality however, 3D extended sound sources[35] are difficult to achieve in 3D audio displays since the distance of virtual sound sources is difficult to control (see 5.4.3).

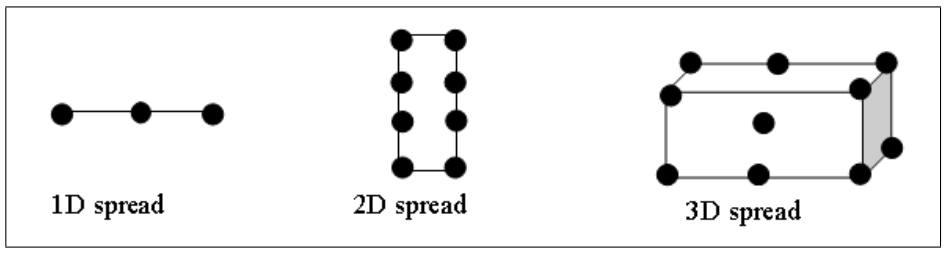


Figure 2.38: Creation of 1D, 2D and 3D broad sound sources using the decorrelated point source method

[34] spatialisation is detailed in 5.4.1
[35] such as wind blowing in trees

2.12.4 Extension and evaluation of the decorrelated point source method

Although described in [Ken95, Sib02], the decorrelated point source technique for rendering sound source extent was not subjectively evaluated. In chapter 4, this thesis presents a thorough perceptual investigation of the present technique through various psychoacoustic experiments and proposes that this technique may also be used to render the apparent shape of sound sources.

2.12.5 Obtaining decorrelated signals

To obtain a set of decorrelated signals that are then fed to the discrete sound sources, a straight forward method consists of sampling a broad natural sound source with a microphone array (Fig. 2.39). Then, to render the broad sound source in a 3D audio scene, the microphone signals are spatialised at their respective positions in the microphone array. This recording technique has been used in the context of the Wave Field Synthesis (WFS) [VB99, Boo95, Ber88] spatialisation technique; the microphone array approach is expensive and requires a relatively large number of audio channels however.

An alternative method is to apply *decorrelation* on the monaural source signal so as to obtain a set of decorrelated signals; these are then fed to discrete sound sources (Fig. 2.40). This technique has the advantage that any monaural recording can be used to produce broad sound sources. Several signal decorrelation techniques are now reviewed in section 2.13.

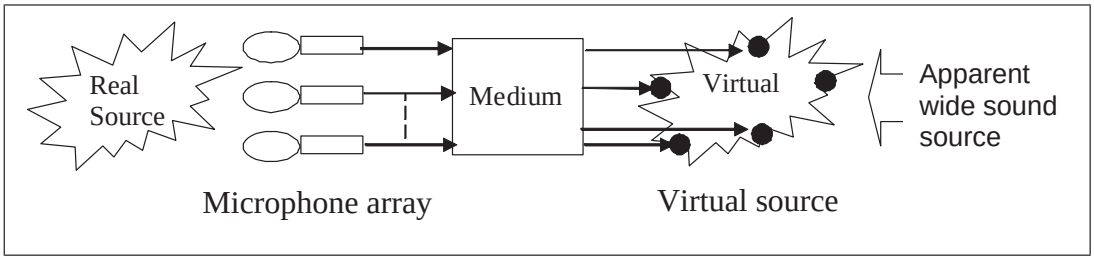


Figure 2.39: Capture and reproduction of the extent of a natural sound source via a microphone array

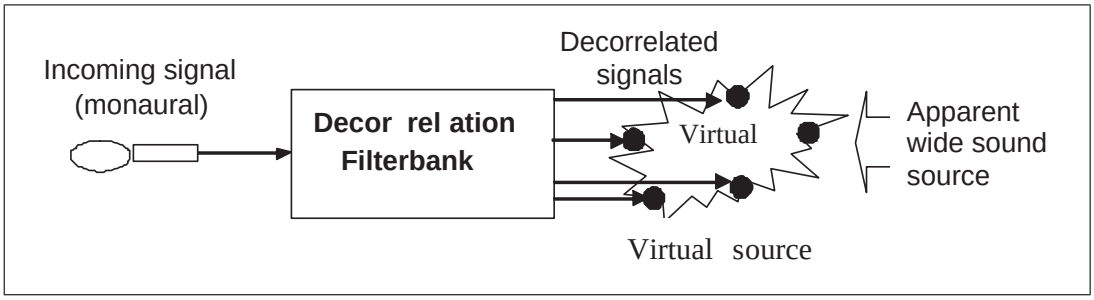


Figure 2.40: Capture and reproduction of the extent of a natural sound source via a single microphone and a decorrelation filterbank

2.13 Signal decorrelation techniques

This section reviews several signal decorrelation techniques that can be used for the purpose of sound source extent and shape rendering (section 2.12). The aim of decorrelation is to generate, from a monaural input signal, a set of statistically orthogonal signal replicas, which taken individually, are perceptually identical to the input signal. Decorrelation strength can be measured using the cross-correlation coefficient and the coherence matrix which are defined in Annex 7.1.

In addition to being used for rendering sound source extent, signal decorrelation techniques are also employed in a number of other applications, and it is useful to review this literature; these applications are:

- in multi-channel acoustic echo cancellers [LI02, Ali98] to increase the convergence speed of adaptive filters

- in blind source separation [FP02]
- in speech and audio coding [Yan00]
- in sound reinforcement systems, to defeat the precedence effect[36] [KWM93, KKFW99]

Different techniques for achieving signal decorrelation are now reviewed. Full-band and time invariant decorrelation methods are first studied.

2.13.1 Time delay

The simplest form of signal decorrelation consists of introducing a small time delay between copies of the input signal [Vag01]; this is depicted in Fig. 2.41. Although simple and cheap to implement, this technique however, has a major drawback when used to render sound source extent with several point sources. Indeed, the delays introduced between the output signals create comb-filtering[37] effects which modify the timbre of the sound source signal in an unpleasant manner. Comb filtering effects attenuate parts of the source spectral content and, due to less energy in the signal spectrum, this reduces the perceived extent of the sound source (see section 2.6). Comb filtering may also negatively affect perceived extent as colouration tend to produce unnatural sounding and 'tinny' sources. Thus, delay based decorrelation is not advisable to be used for the task of sound source extent rendering.

Delay based decorrelation is, however, suitable to create spatially extended sound sources when large delays (several seconds) are used. Large delays, however, cannot be used with all signals such as music and speech but can be used for constant and periodic signals such as waves breaking on a beach, the sound of rain etc. In this case, the delay should be long enough as to avoid echoes and identification by listeners of the signal replicas using memory.

[36] the precedence effect is reviewed in section 2.9.1

[37] If delayed signal replicas are roughly within 50 ms of the original signal otherwise an echo is heard

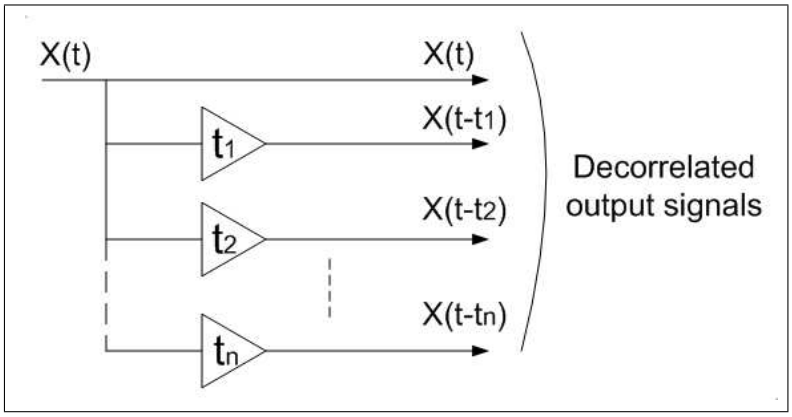


Figure 2.41: Obtaining decorrelated signals by delaying an input signal

2.13.2 Fixed FIR all-pass filters

A more advanced method to obtain a set of decorrelated signals is to perform convolutions of the input signal with all-pass FIR filters having random, noise-like, phase responses. The resulting random phases of the output signals, in turn, produce statistically orthogonal signals [Ken95, Ken94]. This technique is depicted in Fig. 2.42. Due to the insensitivity of the binaural system to phase variations of a signal [PS69] and due to the preservation of the input signal spectrum (all-pass response), the obtained output signals are perceptually identical but statistically orthogonal (i.e. decorrelated). The frequency and phase responses of a 256-tap all-pass FIR decorrelation filter are shown in Fig. 2.43 and its impulse response is plotted in Fig. 2.44.

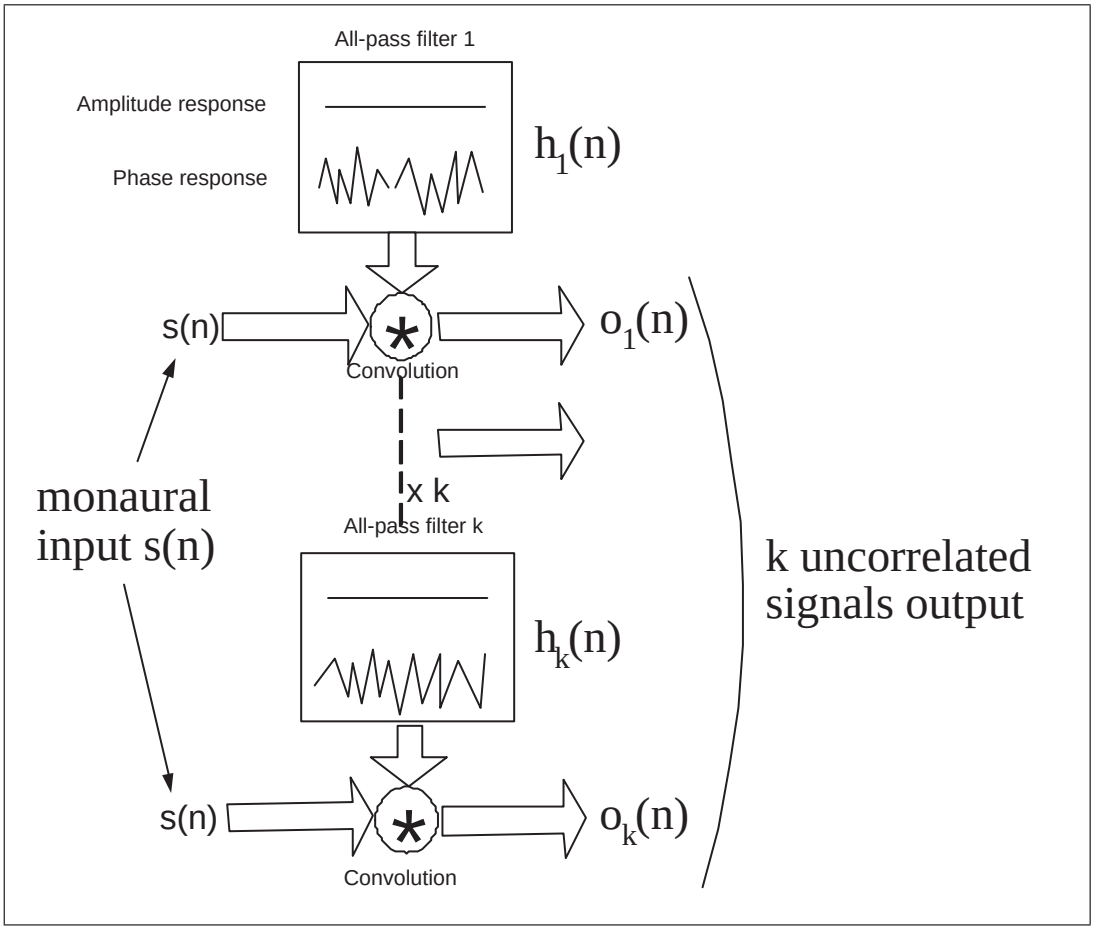


Figure 2.42: Decorrelation filterbank to create several uncorrelated replicas of a monaural signal

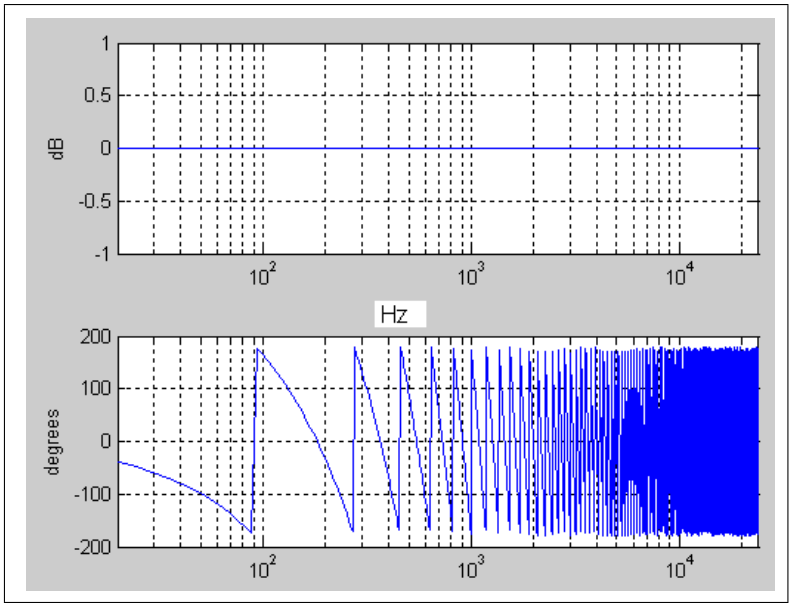


Figure 2.43: Frequency and phase response of an all-pass FIR decorrelation filter

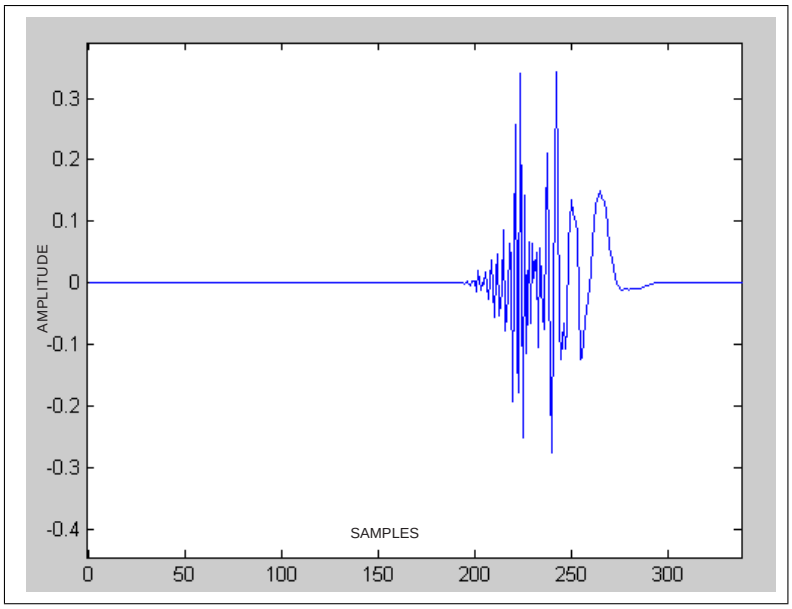


Figure 2.44: Impulse response of an all-pass FIR decorrelation filter

To obtain a FIR decorrelation filter with phase and frequency responses similar to that depicted in Fig. 2.43, a technique known as the frequency response sampling method [Ken95, Mit03] can be used. This technique is summarised in Fig. 2.45 and consists of creating a vector A representing the filter magnitude response which is filled with ones (all pass response) and a vector B representing the filter phase response which is filled with a random signal ranging between -180 and +180 degrees. The frequency response of the FIR filter in complex vector form is then described by the vector H_f:

$$H_f = A.(cos(B) + j * sin(B)) \tag{2.9}$$

The coefficients of the all-pass FIR filter are then obtained by transforming H_f into the time domain using an inverse Fast Fourier Transform:

$$H_n = FFT^{-1}(H_f) \tag{2.10}$$

There are, however, several problems with the fixed FIR decorrelation technique. First, if too few taps are used for the all-pass filters, little decorrelation is achieved due to insufficient 'phase shuffling'. Another issue when too few taps are used, is that the sampling of the filter magnitude response is less accurate and the amplitude responses of the obtained decorrelation filters (after the inverse FFT operation) are not perfectly flat; this introduces coloration on the decorrelated signals. On the other hand, if too many taps are used, the signal is smeared in the time domain; this is problematic with signals having fast onsets (e.g. percussive sounds). In order to limit this effect, Kendal [Ken94] recommendes to use filter lengths shorter than 20ms[38]. The author, however, recommends to use even shorter filters of less than 10ms since temporal smearing effects were noticeable (and distracting) for filter longer than 10ms. The tap-length of the FIR decorrelation filters must therefore be chosen as a compromise between decorrelation strength and limitation of the temporal smearing effects. The

[38]That is, 882 FIR taps at a 44.1 kHz sampling frequency

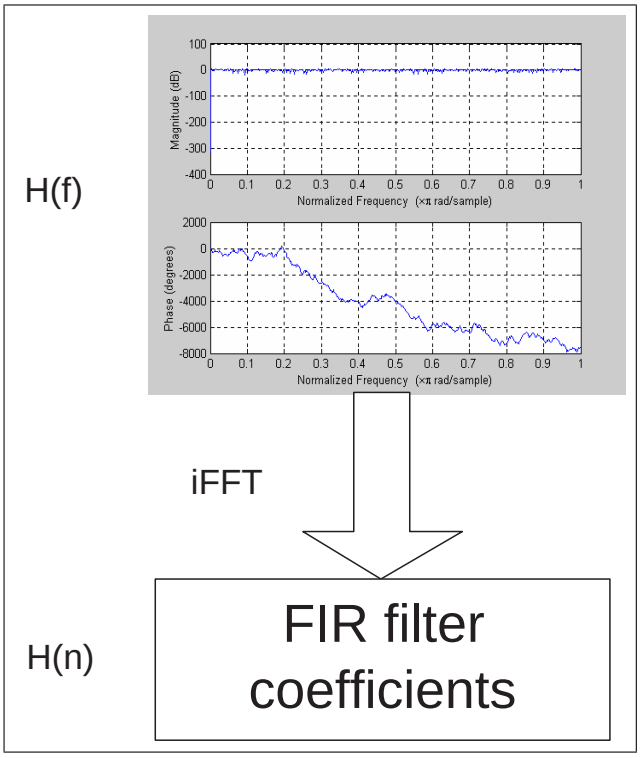


Figure 2.45: Obtaining an all-pass FIR decorrelation filter via artificial magnitude and phase response construction and inverse Fast Fourier Transform

author found that FIR decorrelation filters with 256 taps were a good compromise. Further remarks on fixed decorrelation can be found in section 2.13.5.

2.13.3 Fixed IIR all-pass filters

IIR decorrelation all-pass filters can be devised using the following well known property of IIR all pass filters [Mit03]:

$$H(z) = \frac{a_N + a_{N-1}z^{-1} + a_{N-2}z^{-2} + \ldots + z^{-N}}{1 + a_1 z^{-1} + a_2 z^{-2} + \ldots + a_N z^{-N}} \tag{2.11}$$

Equation 2.11 shows that zeros and poles of all-pass filters are complex conjugates

since the polynomial coefficients at the numerator are running in reverse order to that of the denominator. To create decorrelating IIR all-pass filters with random phase responses, a technique consists of randomly placing a number of poles within the unity circle (to guaranty filter stability)[Ken94]. Then, by finding the polynomial coefficients of Equation 2.11 for the randomly placed poles, the a_k filter coefficients are obtained. The advantage of the IIR decorrelation technique is that a lower filter order is required compared to the FIR approach to achieve the same amount of decorrelation [Ken95]. The author implemented the described IIR decorrelation filter design technique in a Matlab script which can be found in Annex 7.2. Further remarks on fixed decorrelation can be found in section 2.13.5.

2.13.4 Feedback Delay Networks

Feedback Delay Networks (FDN) is a technique originally developed by Jot [Jot97, VVHK97] for simulating room reverberation. The FDN technique in its simplest form, uses delays, amplifying gains and a circulant matrix (Fig. 2.46). More details of the FDN architecture can be found in [GT02, RS97].

Rocchesso [ZA02] proposed that FDNs could be used to achieve signal decorrelation by selecting parameters that produce an all-pass filter behaviour rather than reverberation. In his proposal, Rocchesso however, does not specifies if this technique could be used to produce a large number of decorrelated signals.

2.13.5 Remarks on fixed decorrelation

One issue with fixed decorrelation techniques (FIR or IIR) is that they can only produce a relatively small number[39] of decorrelated signals, because a high correlation value will eventually occur between a particular pair of output signals; this is due to the finite length of the filters used. Therefore, the claim made by Kendal [Ken95] who declared that an unlimited number of decorrelated signals could be obtained from

[39]From the author's experience, under ten decorrelated signals

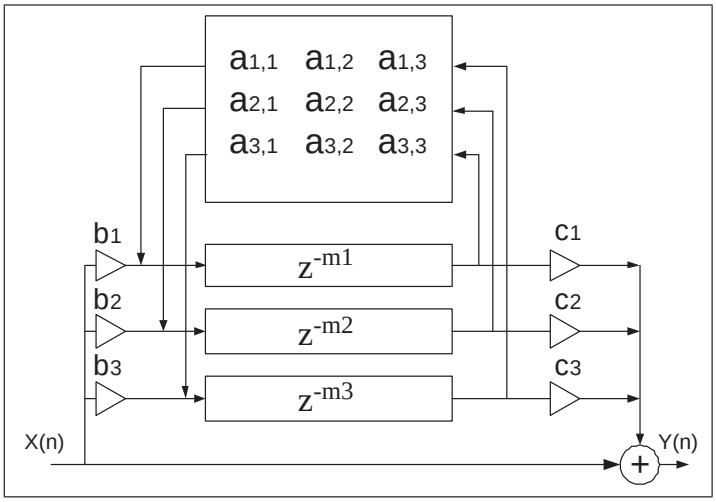


Figure 2.46: Architecture of an order 3 feedback delay network

an all-pass decorrelation filterbank does not seem possible. However, the number of decorrelated signals obtained with fixed decorrelation is usually sufficient to render sound source extent and shape (section 2.12).

Another problem with fixed decorrelation is that the input signal to decorrelate usually exhibits a time varying spectrum (e.g. music or speech) and therefore, phase alterations (and decorrelation) produced by the decorrelation filters only apply at frequencies at which signal spectrum is present. This produces decorrelation that is positively influenced by the richness of the input signal spectrum and furthermore, that is temporally varying with the signal spectrum. In order to obtain further signals, and to achieve decorrelation that is less sensitive to the input signal, dynamic decorrelation is used.

2.13.6 Dynamic decorrelation

Instead of using constant all-pass filters as described in section 2.13.2 and 2.13.3, dynamic decorrelation consists of periodically updating the random phase responses of the all-pass filters on a time frame basis. FIR or IIR lattice filter structures [Mit03] are best suited for this task thanks to their resilience to instabilities that may occur

during frequent filter coefficient updates.

Kendal [Ken95] states that "Dynamic decorrelation produces a special effect akin to the sound of an environment with moving reflective surfaces or moving sound sources, such as the movement of leaves and branches in a forest or the movement of a crowd within an auditorium. Dynamic decorrelation imparts a quality of liveliness to a sound field that is missing in the (fixed) FIR implementation". The advantage of dynamic decorrelation over fixed decorrelation is that a higher number of uncorrelated output signals can be obtained [Ken95], however, dynamic decorrelation can lead to listening fatigue due to the constant changes of the decorrelation filter phase responses. The perceptual effects of dynamic decorrelation are studied in a psychoacoustic experiment described in section 4.9. The author implemented a dynamic decorrelation filter in Matlab, details can be found in Annex 7.3.

2.13.7 Frequency varying decorrelation

So far, decorrelation has only been applied to the full signal spectrum. Frequency varying decorrelation is a technique whereby decorrelation is performed only on selected sections of the input signal spectrum [PB04b]. Combined with the method described in section 2.12 for rendering sound source extent, frequency varying decorrelation can lead to interesting effects where the spatial extent of a sound source varies with frequency. In informal listening tests carried out by the author, using this technique, variations of source extent with frequency were clearly noticeable.

The diagram of a sub-band decorrelator is depicted in Fig. 2.47 and performs as follows: the input signal is first split into sub-bands using a decomposition filterbank made of brick wall low-pass, band-pass and high-pass filters. Each sub-band signal is then decorrelated using a normal FIR (section 2.13.2) or IIR (section 2.13.3) decorrelation technique. The different partially decorrelated signal sub-bands are added together to form the final set of sub-band decorrelated signals.

Cross-fader modules are then used to control the amount of decorrelation in each frequency band using a correlation factor k. This works by re-injecting some common

sub-band signal into each decorrelated signal. For example, if total decorrelation is wanted, k equals zero, then no common signal is injected. If k equals one, the cross-fader outputs only the common signal and no decorrelated signal, therefore the correlation coefficient is one. It is also possible to set k to any intermediate correlation value. A constant power cross-fading [Wes98] technique is preferable so that no change in signal level can be observed when k is varied.

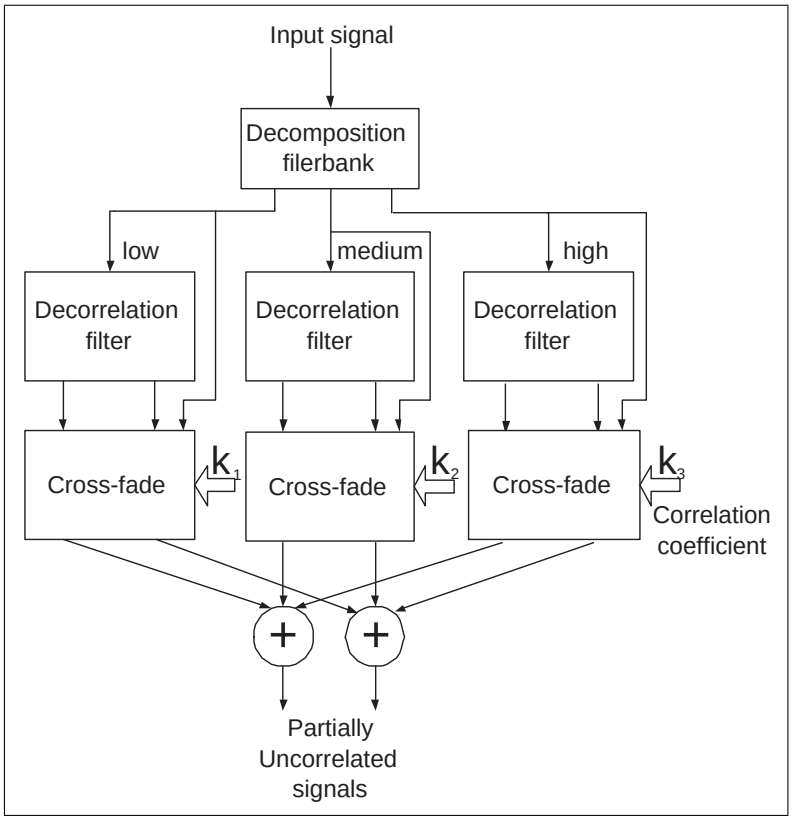


Figure 2.47: Principle of a sub-band decorrelator

2.13.8 Time varying decorrelation

Decorrelation can also be produced so that it is varying with time. This is achieved by using a cross-fader module that is controlled by a time varying function $k(t)$. Similarly to the sub-band decorrelator described in section 2.13.7, the cross-fader

module is used to control the amount of common signal and decorrelated signals into the output signals, and when $k(t)$ is varied, this creates decorrelation with a temporal change in decorrelation strength. It is to be noted that this technique is not equivalent to dynamic decorrelation (section 2.13.6) which consisted of updating the decorrelation filter coefficients. The diagram of a time-varying decorrelator is shown in Fig. 2.48. In section 4.9 time varying decorrelation is subjectively tested in an experiment.

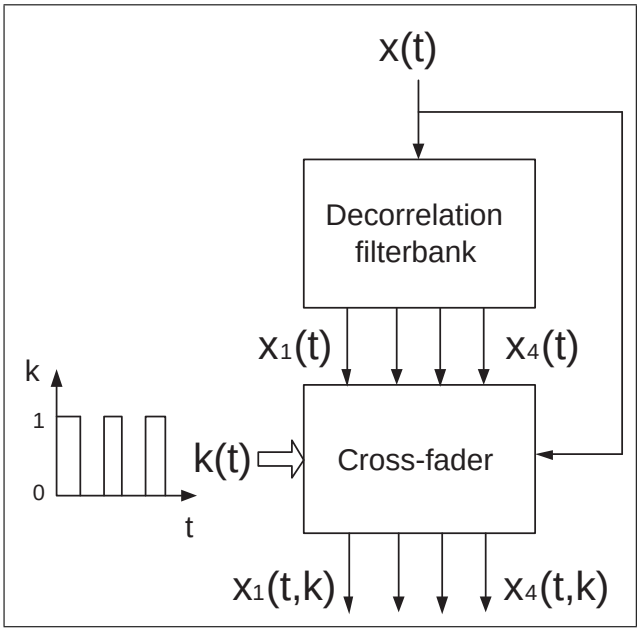


Figure 2.48: Principle of a time-varying decorrelator

2.13.9 Other decorrelation techniques

Other decorrelation techniques besides delay and all-pass filtering exist (see [LI02] for an overview), these are often used in echo-cancellation systems. However, these techniques are best avoided for rendering sound source extent because they either degrade the signal (artificial introduction of noise and distortion), create disturbing phasing effects (Hilbert transform based techniques) or do not generate a high enough number of decorrelated signals.

2.13.10 Summary of source extent rendering techniques

Several techniques were described and compared to control the extent or sound sources. Emphasis was placed on a method that uses decorrelated point sources to render 1, 2 or 3D extended sound sources; this technique is subjectively tested in several experiments described in chapter 4. Several signal decorrelation techniques were then described; these can be used for the purpose of rendering spatially extended sound sources, using the decorrelated point source technique.

2.14 General summary

Channel and object oriented techniques for transmitting 3D audio content were reviewed. The advantages of the object-oriented approach in terms of scalability and interactivity were explained. Shortcomings of the scene graph model used in object-oriented standards such as VRML and MPEG-4 were then highlighted. In chapter 3 is presented an alternative 3D audio scene description scheme that addresses these problems. The practical implementation of the scheme in a novel 3D audio rendering system is then described in chapter 5.

Spatial auditory percepts were then reviewed. A particular focus was placed on the perception of sound source extent in the presence of one and multiple sound sources. Techniques for artificially rendering sound source extent in 3D audio displays were then described. A technique which consists of using several decorrelated sound sources to produce artificial sound source extent was reviewed. In chapter 4, this technique is perceptually evaluated.

Chapter 3

Novel object-oriented approach for describing 3D audio scenes using XML

3.1 Introduction

Non-object and object oriented methods for transmitting and storing 3D audio scenes were reviewed in section 2.2. It was shown that non-object or channel oriented techniques do not allow interactivity with the scene and usually impose restrictions on the rendering terminal, such as, for instance, the obligation to use the correct number and placement of speakers. In contrast, the object oriented approach provides a higher level of abstraction so that 3D audio scenes can be described independently from a particular rendering configuration, allowing the transmission of the same scene to a wide range of terminals (e.g. headphones, two speakers, custom speaker array etc). Besides, object oriented techniques allow interactivity and modification of the 3D audio scenes by the end user.

This chapter presents a novel XML based object-oriented scheme for describing animated 3D audio scenes. The scheme, called XML3DAUDIO, is novel in that it does not follow a traditional scene graph model that is used in standards such

as VRML, X3D and MPEG-4 AudioBIFS[1]. Instead, XML3DAUDIO follows a new scene orchestra and score approach. It is shown in this chapter that, unlike the scene graph model, this new approach allows the scene content data to be separated from the scene temporal data. This, in turn, simplifies and clarifies the scene description. It is also shown that this approach allows centralisation of the description of the scene temporal behaviour in the scene score; this allows easier modification and re-authoring of the scene description. The advantages of the new scene orchestra and score approach over the scene graph approach for describing 3D audio scenes are highlighted in several 3D audio scene examples.

The novel XML3DAUDIO scheme presented in this chapter was initially developed to be used as an ad-hoc format for describing and rendering 3D audio scenes; its practical implementation in a 3D audio rendering system is described in chapter 5. Existing standards (i.e. VRML, X3D and MPEG-4) were discarded for this task as they are either too simple and lack description capabilities (VRML, X3D) or are too complex (MPEG-4 AudioBIFS). For instance, MPEG-4 Advanced AudioBIFS has not, to this date[2], been fully implemented in a MPEG-4 compliant player due to the high complexity of the standard.

While benefiting from a simpler and viable format for fully describing and rendering animated 3D audio scenes, XML3DAUDIO encompasses state of the art 3D audio description features[3] found in MPEG-4 Advanced AudioBIFS (AABIFS). In addition, XML3DAUDIO provides novel features that are currently unavailable in MPEG-4 AABIFS. These include: the ability to algorithmically compose 3D audio scenes, the ability to compose hybrid scenes[4] and the ability to describe the spatial extent of sound sources.

[1]These standards were reviewed in chapter 2

[2]That is, five years after reaching the Final Draft International Standard (FDIS) status

[3]Feature examples: directivity of sound sources, reflectivity of acoustic material, definition of room reverberation etc.

[4]Hybrid 3D audio scenes were defined in 2.2

The design aims of XML3DAUDIO are first listed. The different areas of description required to fully define 3D audio scenes are then identified. The new scene orchestra and score approach is then explained. The formats of the scene orchestra and scene score are then detailed. Follows the description of the different scene objects of the novel scheme that are used to describe the content of 3D audio scenes. An evaluation of the new scheme is then given, to do so, the scheme features are compared against the 3D audio description capabilities of MPEG-4 AudioBIFS. The new scene orchestra and score approach used by the novel scheme is then compared in terms of efficiency and clarity against the scene graph model. Finally, an alternate use of the new scheme as a meta-data annotation scheme for describing 3D audio content is explained.

3.2 XML3DAUDIO: A new 3D audio scene description scheme

3.2.1 Design philosophy and aims of XML3DAUDIO

The purpose of XML3DAUDIO is to provide a viable framework for thoroughly describing animated 3D audio scenes in an object oriented way so that, from a scene description and sound resources (i.e. sound files or streams), a complete 3D audio scene can be rendered. The design criteria of XML3DAUDIO are now listed.

- Simplify as much as possible the syntactic structure of the 3D audio scene descriptions, while offering advanced features for describing 3D audio scenes. Simplification of the scene description was intended to allow easy authoring of 3D audio scenes and to improve scene parsing efficiency at the rendering stage. It is explained in section 3.3 how the selected orchestra and score model used to develop the scheme simplifies scene description. A comparison of the novel

scheme with the scene graph model in terms of scene description simplicity is given in section 3.4.2.

- Provide state of the art descriptors to define 3D audio scenes. To do so, 3D audio scenes can be described in a physical way (e.g. sound source extent, sound source directivity, material reflectivity etc) and in a perceptual way (reverberation parameters such as warmth etc), this dual approach is also used in the MPEG-4 AABIFS standard (see 2.4.2).

- Centralise the scene temporal behaviour description (e.g. playing times of sound sources, change of object parameters over time etc) and separate it from the scene content description. This structure, is turn, permits easy authoring and re-authoring of 3D audio scenes. Centralisation of the scene temporal data contrasts with the scene graph approach where scene content and scene timing descriptions are aggregated; this results in complex and tangled scene graphs and the spreading of the scene temporal information throughout the scene graph. Advantages of separating scene content and scene timing descriptions are further highlighted in section 3.4.2.

- Do not define complex interactive behaviours such as found in VRML or MPEG-4 (section 2.4.1). This was decided since interactivity is mainly useful in virtual reality scenes where both visual and auditive feedback are present. Even though interactivity is not implicitly described in the scene description, users may still interact with the scene by modifying the parameters of objects at the rendering side, such as for instance, the position of sound sources, playing or stoping sound sources etc.

- Allow definition of parent-child relationships between objects so that complex objects[5] can be manipulated as single objects, this feature is equivalent to *Group* and *Transform* nodes of VRML and MPEG-4 BIFS.

[5]Which will be called macro-objects in the rest of this chapter

- Remain independent from any particular 3D audio format, channel and speaker configuration or 3D sound API; this allows the scheme to be used in a wide range of use case scenarios. For example, the scheme can alternatively be used as a meta-data annotation scheme for 3D audio (see 3.5).

- Allow complex scene authoring commands to be issued by scene authors to compose 3D audio scenes algorithmically. For example, a 3D audio scene containing a swarm of bees can be described by a single 'swarm' command which results in cloning a number of sound sources, playing at random times and having individual random trajectories. The use of algorithmic composition commands allows complex 3D audio scenes to be described very efficiently. This novel feature has no equivalent in VRML, X3D or MPEG-4 AudioBIFS.

- Allow construction of hybrid 3D audio scenes. Hybrid 3D audio scenes are the combination of natural (i.e. captured) and synthetic 3D audio scenes artificially (composed of spatialised sound sources). To achieve the description of hybrid scenes, XML3DAUDIO allows importing recorded 3D audio scenes in the currently described scene. This feature does not exist in VRML and MPEG-4 and it is shown in section 3.4.3 the reasons as to why it is problematic to devise hybrid 3D audio scenes in MPEG-4 AudioBIFS.

The eXtensible Markup Language (XML) was selected to develop XML3DAUDIO since it is widely used for meta-data and description tasks [Cah01] and that many tools for authoring and processing XML are widely available [Tit04]. Besides, XML is human readable and thus, 3D audio scenes can be authored using a simple text editor. XML3DAUDIO is implemented as an XML schema [Tit04] and therefore, a described 3D audio scene results in an XML file conforming to this schema. Due to the extensible nature of XML, later extensions of XML3DAUDIO are also possible.

3.2.2 3D audio scene description areas

The design of XML3DAUDIO was based on the observation that 3D audio scenes are formed by the interaction between objects such as: the listener, sound sources, reflecting and obstructing objects, reverberant spaces and medium. In addition, these scene objects can exhibit time varying behaviours, such as the playing times of sound sources and the change of object parameters over time (e.g. change in position). Lastly, hierarchical and contextual relationships between objects may exist, such as parent/child relationships and acoustical environments (i.e. objects belonging to the same room).

XML3DAUDIO describes complete 3D audio scenes by three means: the description of scene content, the description of scene timing and the description of scene hierarchy; these three areas are illustrated in Fig. 3.1. It is explained in section 3.3 that the new scene orchestra and score approach, unlike the scene graph model, permits the clear separation of scene content description from the temporal and hierarchical scene description. The three categories of scene description are now detailed.

Scene content

The scene content is defined by a list of *Elementary* objects present in the 3D audio scene. Elementary objects include the listener (i.e. the reference point), sound sources, reflective surfaces (reflecting or obstructing sounds), 3D audio recordings (used to create hybrid scenes), rooms (describe reverberation) and medium (describes propagation properties). The description of the scene content is performed in the scene orchestra (section 3.3.4) and elementary objects are detailed in section 3.3.6.

Scene timing

The scene *timing* descriptors then describe the scene temporal behaviour such as the playing times of sound sources and change of object parameters over time (e.g. their positions). The description of the scene timing is performed in the scene score

(section 3.3.5).

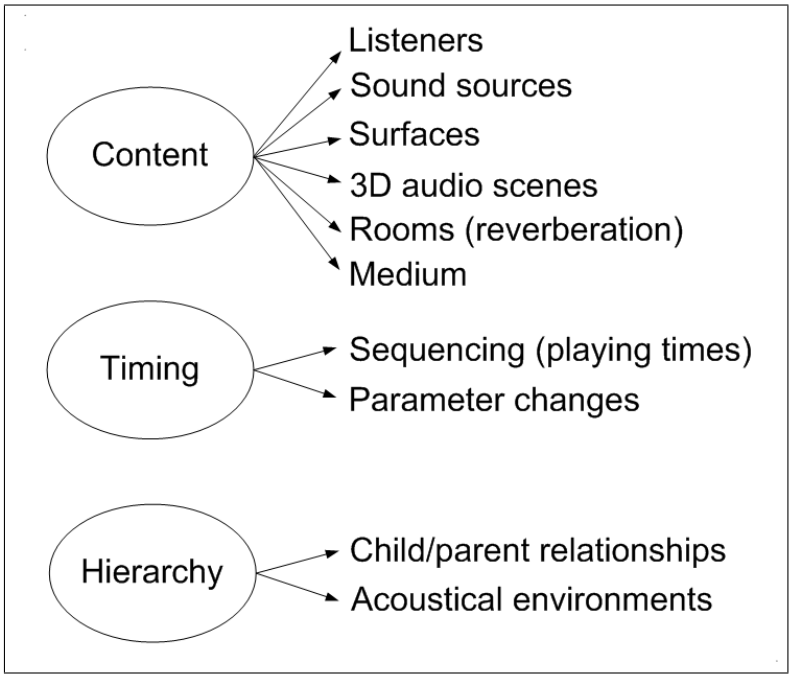


Figure 3.1: Illustration of the three description categories to describe 3D audio scenes

Scene hierarchy

The scene *hierarchy* descriptors define the relationships between objects. This addresses the fact that complex sound objects can be composed of several elementary objects. For example, a complex object such as a car has a reflective body and emits multiple sounds: exhaust, engine noise, tyre friction, horn etc. Complex objects are called macro-objects and their description in the novel scheme is explained in section 3.3.6. The *hierarchy* descriptors are also used to define acoustical environments, these are groups of objects that only interact between each other (e.g. objects belonging to the same room). Environments are further explained in section 3.3.

3.3 The scene orchestra and score approach

XML3DAUDIO follows a scene orchestra and score approach. The orchestra and score model is inspired by the sound synthesis programming language CSound [Bou00]. The CSound model was also used in the creation of the Structured Audio Orchestra Language (SAOL) [VS] which is the sound synthesis and processing language of MPEG-4 AudioBIFS[6].

3.3.1 Scene orchestra: content description

In the scene orchestra and score approach, the *scene orchestra* only describes the scene content by listing the objects present in the 3D audio scene (e.g.. sound sources, listener, rooms etc) and describes their intrinsic properties. In the orchestra, no such child/parent relationships between objects are defined (as in the scene graph model), and objects are just described at the same hierarchical level (Fig. 3.2). The hierarchical relationships between objects are instead described in two ways: in the *scene score* or by importing *macro-objects* in the *scene orchestra*, this is explained in section 3.3.6.

In the scene orchestra, each object and macro-object has a unique ID which is used by the scene score to address the objects (Fig. 3.2). The format of the *scene orchestra* is detailed in section 3.3.4.

3.3.2 Scene score: initialisation, timing, composition and hierarchy description

The *scene score* first performs initialisation of the scene, this is used to set the properties of the orchestra objects before the scene is rendered. Scene score initialisation is detailed in section 3.3.5.

[6]SAOL is implemented by the AudioFX node of AudioBIFS

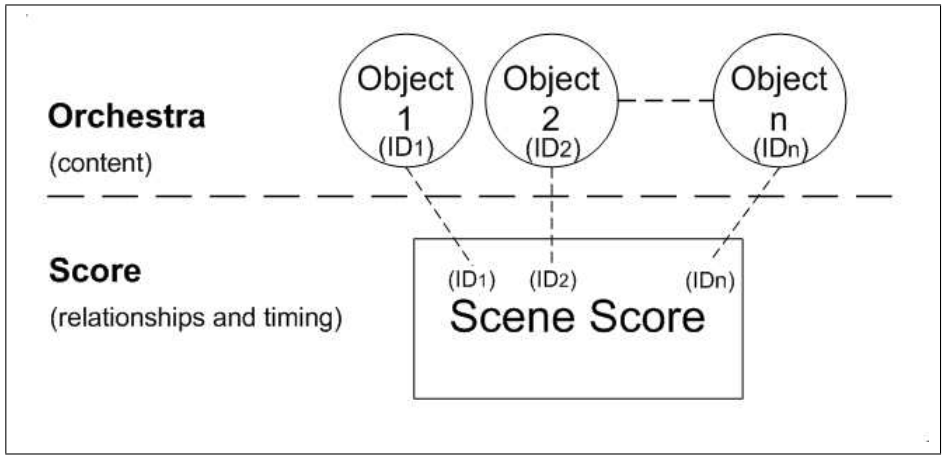


Figure 3.2: Overview of the orchestra and score approach

The *scene score* then describes scene animation and the modifications of object parameters over time (e.g. position of sound sources). The scene score also defines the sequencing and playing times of sound sources. A sound source may be played several times in arbitrary ways by the scene score, this contrasts with the scene graph model where playing times are embedded in the fields of the AudioClip[7] object itself, resulting in a fixed temporal use of the sound source, unless routing mechanisms are used to modify the fields of the AudioClip object, adding complexity to the scene description.

The *scene score* is also used to describe eventual parent/child relationships between objects. This grouping mechanism is useful when animating complex objects, so that a single animation command is required in the score instead of animating individual objects of the macro-object. In the scene graph approach, grouping of objects is performed with *Group* and *Transform* nodes (section 2.4.1).

The scene score may also be used to algorithmically compose 3D audio scenes. Using special composition commands[8] in the scene score, complex 3D audio scenes can be quickly composed. This feature is unique to XML3DAUDIO and is described in section 3.3.5. Commands specified in the scene score may also be used to alter the

[7]See section 2.4.1
[8]That are called opcodes

scene dynamically, such as instantiating and killing objects etc. This feature can be compared to the *BIFS-Commands* mechanism of MPEG-4 BIFS (section 2.4.2).

Lastly, the scene score can be used to define acoustical environments. An environment is a mechanism used to group several objects into a region where they only interact between each other, for instance, sound objects belonging to the same room. This feature is useful to describe 3D audio scenes that have multiple acoustical environments (e.g. different rooms). This feature is also present in MPEG-4 AudioBIFS and is implemented by the *AcousticScene* node [PE02]. The format of the *scene score* is defined in section 3.3.5.

3.3.3 Benefits of the scene orchestra and score approach

The scene orchestra and score approach was selected to develop XML3DAUDIO since it clearly separates the scene content data from the scene temporal and structural data. Compared to the scene graph model which combines content and timing data in complex ways (section 2.4.1), this approach allows a simpler and more organised scene description. This, in turn, allows a centralised description of the 3D audio scene temporal behaviour which can be easily re-authored and more quickly processed by a scene render. These claims are highlighted in an evaluation of XML3DAUDIO against VRML and MPEG-4 in section 3.4.2.

The format and structure of the scene orchestra and the scene score are now detailed. This is followed by the description of the objects that can be used in the scene orchestra.

3.3.4 Format of the scene orchestra

The *scene orchestra* simply lists the objects present in the 3D audio scene and describes their intrinsic properties. Setting object properties serves as initialisation of the objects before the start of the scene; scene initialisation may also be performed by the scene score (see 3.3.5).

Fig. 3.3 shows the XML schema of the scene orchestra. The list of objects that can be be used in the 3D audio scene orchestra and their functions are summarised in table 3.1. There can be an unlimited number of instances for each type of object. The orchestra objects and their parameters are detailed in section 3.3.6. Every object described in the scene orchestra has a unique ID. This is used to link the scene objects described in the scene orchestra with their temporal descriptors defined in the scene score. The straight forward nature of the scene orchestra format allows fast parsing of the 3D audio scene content by a scene renderer. For a human, the scene orchestra XML data may be read directly without the need of particular tools (an XML scene example is given in section 3.3.7). In comparison, 3D audio scenes described in VRML or MPEG-4 XMT[9] can appear as complex scene graphs having deeply nested objects. This complicates the task of the scene parser/renderer and reduces readability of the textual scene description by humans.

[9]That is, MPEG-4 AudioBIFS described in XML syntax (section 2.4.2)

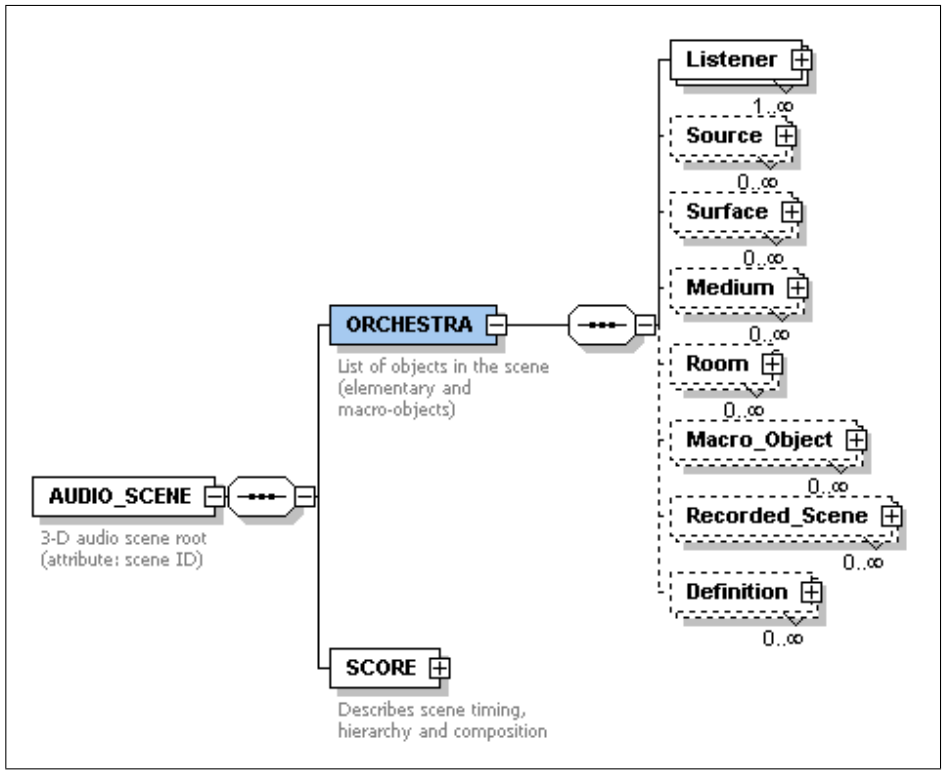


Figure 3.3: Format of the scene orchestra

Object	Function
Listener	Reference position in the 3D audio scene
Source	Sound emitting object
Surface	Reflective and obstructing object
Medium	Defines sound propagation properties
Room	Defines reverberation
Macro_Object	Imports complex object in 3D audio scene
Recorded_Scene	Imports 3D audio recordings (used for hybrid 3D audio scenes)
Definitions	Defines properties of objects which are commonly used

Table 3.1: List of orchestra objects

3.3.5 Format of the scene score

The *scene score*, held in a separate section of the XML 3D audio scene description contains two parts: the *initialisation score* and the *performance score*.

The XML schema of the *scene score* is shown in Fig. 3.4. The *initialisation* and *performance score* are composed of *lines* of score. Each line of score contains a single command that is applied to one or several objects. The formats of the line of initialisation and performance score are depicted in Fig. 3.5.

Initialisation score

The *initialisation score* is used to set the initial states of objects before the start of the scene and to describe possible parent/child relationships between objects defined in the scene orchestra.

A line of initialisation score has a command field containing an *Opcode* (i.e. the command), one or several *Object* fields, containing the IDs of objects that are to be affected by the command (i.e. the operands) and one or several *Parameter* fields containing parameters for the command; the format of the initialisation score lines is depicted in Fig. 3.5.

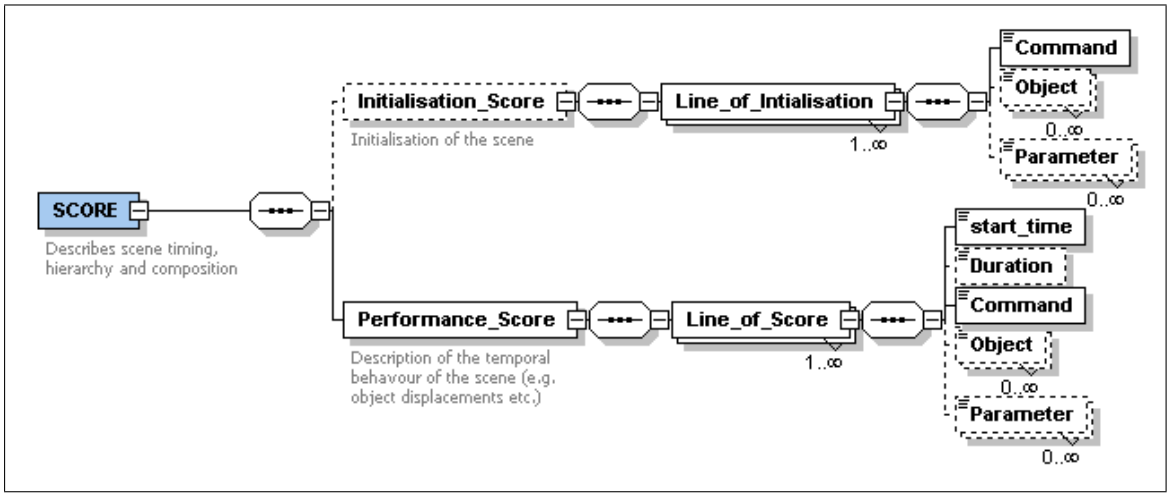


Figure 3.4: Scene score format

Examples of lines of initialisation score are shown in table 3.2 below. The first line of score groups three sound source objects into a new group called 'groupname1'. The second line of initialisation score then places the newly created group object to a point in space at coordinates (-5,0,5).

Command	**Object(s)**	**Parameters**
Group	source1, source2, source3	groupname1
Move_to	groupname1	-5,0,5

Table 3.2: Examples of two initialisation score lines

Initialisation of object properties can also be performed in the scene orchestra as explained in section 3.3.4 above, however the initialisation score overrides initialisation that may have been done in the scene orchestra. The separation of the initialisation score from the performance score allows to easily re-author scenes with different initial states.

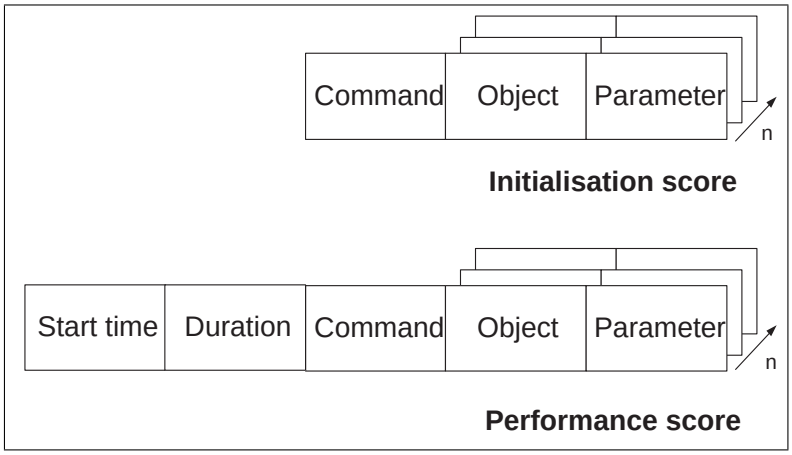


Figure 3.5: Formats of the lines of initialisation and performance score

Performance score

The *performance score* describes the scheduling of the scene (i.e. when sound sources are played), the temporal changes of object parameters (e.g. object trajectories) and hierarchical relationships that are defined *during* the course of the scene. The performance score is also used to algorithmically compose 3D audio scenes using special composition commands that are listed in Fig. 3.6.

A line of performance score has, in addition to a line of initialisation score, a *start time* and a *duration* field; this is depicted in Fig. 3.5. The *start time* defines at what time from the start of the scene ($t = 0$) the action should be undertaken and the *duration* field specifies the action duration, if necessary. Examples of lines of performance score are shown in table 3.3. The first line of score example moves the object 'source1' from its current location to a point in space with coordinates (0,5,-3); this action starts at ($t = 0$) and lasts for five seconds. The scene renderer is responsible for the interpolation of the source position and at $t = 5s$ the *source1* object reaches the target position (0,5,-3).

The second line of score example defines the play time ($t = 5s$) and stop time ($t = 10s$) of the *source1* object. The *loop* parameter specifies that if the duration of

the sound file used by *source1* is shorter than (*stopTime -startTime*), it is looped.

The third line of score example illustrates a complex composition command. Using the 'swarm' command, the *source1* object is cloned at $t = 10s$ into 25 copies, the clones are then randomly distributed in a cube with dimensions $10 \times 10 \times 10m$. Finally, the swarm of sound sources are moved randomly at 4m/s. This example illustrates how complex 3D audio scenes can be devised with only a few lines of scene score.

Start time	**Duration**	**Command**	**Object(s)**	**Parameters**
0	5	Move_To	source1	0,5,-3
5	10	Play	source1	loop
10		Swarm	source1	10,10,10,25,4

Table 3.3: Examples of performance score lines

The different commands that can be used in the initialisation score and performance score are now described.

Score commands

The list of current commands that can be used in the scene score is shown in Fig. 3.6; these are sorted in categories: *scene update commands* that are used to modify objects in the scene, *environment commands* to create virtual acoustic environments, *Object management commands* to dynamically create or delete objects of the scene, *scene hierarchy commands* to define child/parent relationships between objects and to create macro-objects, and finally *scene composition commands* to algorithmically compose 3D audio scenes.

Benefits of the scene score format

From the scene score format that has been described, it can be seen that complex scene behaviours can be described using simple commands. In addition, these complex scene behaviours do not result in higher syntactic complexity of the scene description.

Another advantage of using an opcode approach for the scene score is that the list of score commands is not exhaustive and so the novel scheme can be further extended with more complex scene commands without modifying its basic structure.

In VRML and MPEG-4 BIFS *script* nodes could be used to create complex behaviours such as achieved with the scene score opcodes. One major difference, however, is that *script* nodes cannot instantiate other objects but can only route events to modify object parameters in the scene graph [PE02]. This limits composition features and so, a *script* node would be unable to achieve an equivalence of the 'swarm' command that has been exemplified above. Secondly, *script* nodes require that the script code is defined in the fields of the node itself. Thus, it may be difficult to reauthor a scene graph that contains scripts, since it must first be reversed engineered. In comparison, using XML3AUDIO, the opcodes defined in the scene score clearly indicate the original scene author intentions and thus can be easily modified.

Lastly, the scene score centralises all the temporal and hierarchical scene information in a distinct section of the scheme. Therefore, scenes that are described using XML3DAUDIO may be easily modified. The benefits of the novel scheme approach are further exemplified and compared with the scene graph approach in section 3.4.2.

Command	Behaviour	Parameters
Scene update commands		
Play	Play sound source	Source ID, loop mode
Stop	Stop sound source	Source ID, stop mode
Move	Move object to new location	Object ID, cartesian coordinates
Rotate	Change orientation of object	Object ID, 3D rotation vector
ChangeParameter	Change any object parameter by interpolation	Object ID, keyframe values
Environment Commands		
Env	Group objects in an environment	Object IDs, environment name
EnvBox	Create a 3D box environment	Center, box dimensions
UnEnv	Ungroup objects from environment	Environment name
Object management Commands		
Create	Instantiate new object	Object type, object name
Kill	Kill object from scene	Object name
Clone	Clone one object into several objects	Object ID, new cloned object IDs
Scene hierarchy commands		
Child	Add children to a node	child object IDs, parent ID
UnChild	Remove children from a node	child object IDs, parent ID
Scene composition commands		
Swarm	Clone a sound source into multiple sources and move them randomly	Source ID, swarm dimensions, number of clones, source speed
RandomTraj	Move an object randomly	Object ID, randomness, speed
CircleTraj	Move an object on a circular trajectory	Object ID, center, diameter, rotations, speed
Bullet	Move an object back and forth on a line	Object ID, start point, end point, speed
...	...	...

Figure 3.6: List of scene score commands

3.3.6 List of scene orchestra objects

The functions and the semantics of the different objects that can be used in the scene orchestra are now detailed.

Listener object

The *listener* object defines the position and orientation of the virtual listener in the virtual 3D audio scene. A 3D audio scene may include several *listener* objects. This allows quick switching between several points of view in the scene and allows for multi-user use case scenarios. Switching points of view can be described in the scene score.

The *listener* object has a position and orientation in a 3D left-handed cartesian space. Positions of other objects such as sound sources are then calculated relative to the current *listener* object so that sound sources, for instance, appear at the correct location to the end user. By assigning a trajectory to the *listener* object from the scene score, it is possible to wander in the 3D audio scene.

If the scheme is to be used to render 3D audio scenes binaurally on headphones, the *listener* object describes the Head Related Transfer Function (HRTF) of a particular listener. Using this data, 3D sound scenes can be rendered optimally to the user because his/her own HRTFs will be used (see section 2.3.1). The head-related transfer functions are described by defining IIR filter coefficients for several Azimuth and Elevation angles and for the left and right ears. FIR filters for the HRTFs may also be described if the denominator coefficients 'aCoeffs' are undefined.

It is also possible to describe the HRTF functions by specifying the frequency responses for the right and left ears for particular Azimuth and Elevation angles. Interpolation is then used to calculate intermediate HRTFs [Beg92a] between points of measurement. The listener HRTFs may also be described externally in a *Definition* object. This is useful when several listeners use the same HRTFs, avoiding the repetition of HRTFs descriptions in the XML 3D audio scene orchestra.

The equivalent of the *listener* object in MPEG-4 AudioBIFS is the *ListeningPoint*

object [PE02]. The *ListeningPoint* object, like the *listener* object, describes orientation and position of the virtual listener in the scene. The *ListeningPoint* object, however, does not describe listener's HRTFs because it was decided by the designers of the MPEG-4 standard that this describes details of the terminal and thus it is non-normative. In XML3DAUDIO however, it was decided to allow the optional description of HRTFs since this allows the use of individualised HRTFs in the context of binaural spatialisation.

The semantics of the *listener* object are shown in Fig. 3.7.

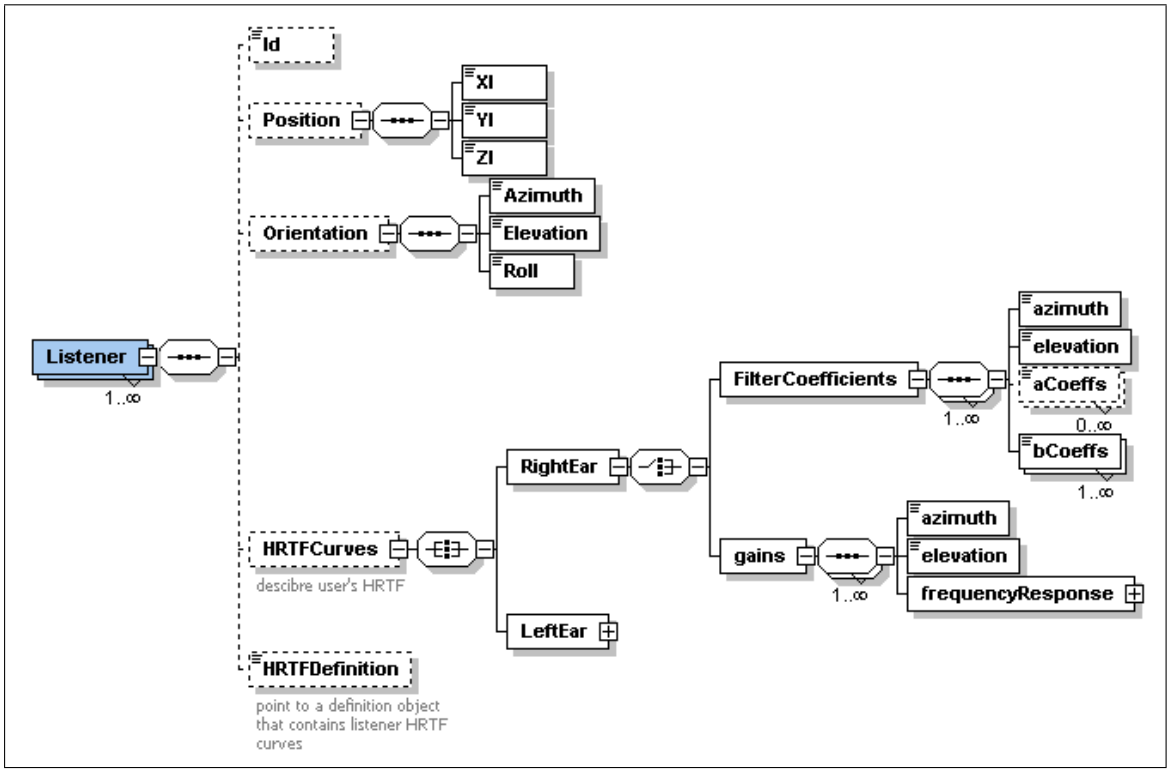


Figure 3.7: Semantics of the *Listener* object

Source object

Source objects are the sound emitting objects in the virtual 3D audio scenes. The *Source* object is used to spatialise monaural sound source signals in 3D audio scenes.

The monaural sound signal emitted by the *Source* object is specified in the URL field of the *Source* object. The sound resource may be a downloadable sound file (e.g. WAV file) or an audio stream.

The *Source* object has a position and orientation relative to the origin of the 3D audio scene. The directivity of the *Source* object can also be described. If source directivity is not specified, the sound source is considered omnidirectional. The directivity of sound sources is described by specifying key attenuation values (in dB) for several source-listener angles, intermediate values are then linearly interpolated. The directivity pattern of sound sources can also be defined at different frequencies since it is known that the directivity of natural sound sources is frequency varying[10] [HST96]. Thus, angular and frequency resolution for describing source directivity is given by the number of entered key values. Fig. 3.8 shows an example directivity pattern described in this manner at two frequencies.

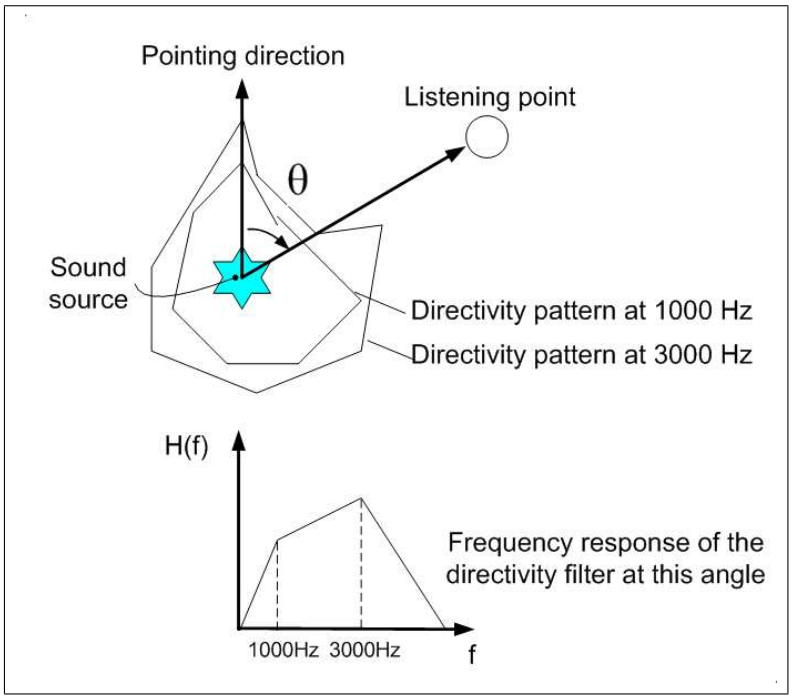


Figure 3.8: Example of sound source directivity described at two frequencies

[10]For instance a cello emits less high frequency at the back than at the front

Sound source directivity may also be described externally in a *Definition* object. This is useful when several sound sources have the same directivity patterns, avoiding description repetition in the scene orchestra.

An alternate method for describing sound source directivity is by means of O-format impulse responses. Instead of describing source directivity point by point, this technique is attractive since it can precisely describe the directivity and spatial extent of sound source from only four impulse responses. Furthermore, the O-format directivity pattern of real sound sources can be measured with microphones. The O-format technique is further explained in section 2.11.4. The O-format impulse responses are described in the *Source* object as an URL pointing to an external 4-channel sound file as this is more efficient than describing impulse responses point by point in XML syntax.

The semantics of the *Source* object then allow for the spatial extent of the sound source to be defined by the dimensions of a 3D box. This permits the description of the following sound sources: point sources (e.g. a flying insect), line sound sources (e.g. a beach front or motorway), 2D sound sources (vibrating panel) and 3D sound sources (a swarm of bees or a waterfall). The extent of sound sources is an important psychoacoustic parameter which is extensively studied in chapter 4.

Finally, the *Source* object may include perceptual parameters describing subjective acoustical properties of the sound source. These parameters are identical to those used in the perceptual approach of MPEG-4 Advanced AudioBIFS (section 2.4.2). The three perceptual parameters are *Presence*, *Brilliance* and *Warmth*. The description of these parameters can be found in [PE02].

The description capabilities of the *Source* object of XML3DAUDIO thus include the capabilities of the MPEG-4 Advanced AudioBIFS *DirectiveSound* object [PE02] in that sound source directivity and perceptual parameters of the sound source can be described. However, the *Source* object offers two novel features over the *DirectiveSound* object: the ability to describe source directivity via O-format impulse responses and

the ability to describe the spatial extent of sound sources. Work presented in this thesis in chapter 4, however, resulted in the addition of sound source extent description capabilities in version 3 of the MPEG-4 AudioBIFS standard. Thus, sound source extent description capabilities will soon be available when MPEG-4 AudioBIFS version 3 reaches FDIS[11] status in 2005.

The semantics of the *Source* object is shown in Fig. 3.9.

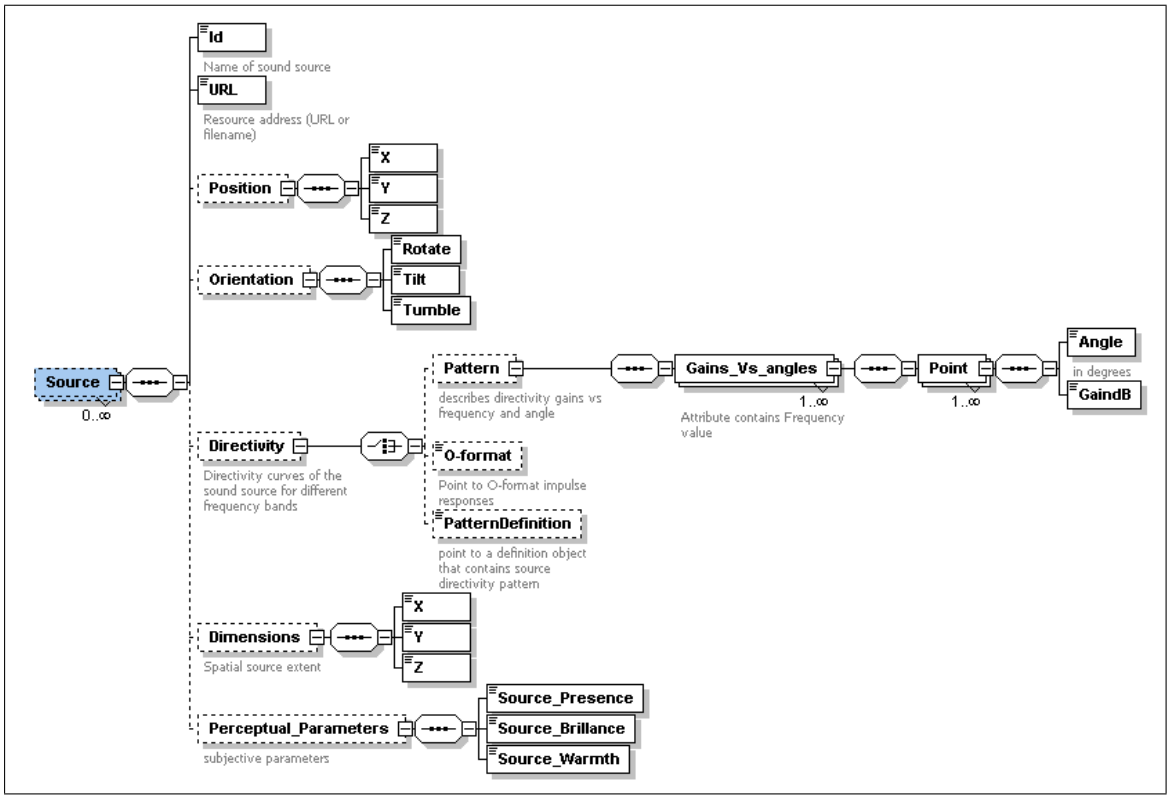


Figure 3.9: Semantics of the *Source* object

[11]Final Draft International Standard

Surface object

Surface objects are used to define objects that obstruct and reflect the sound field in the 3D audio scene. Several *Surface* objects can be used to describe room geometry, this permits the scene renderer to calculate the early room reverberation using a ray tracing or image model algorithm (see 5.4.7). Description of surfaces also allows the scene renderer to calculate obstructions of sound sources by obstacles (such as for instance, to simulate sound sources that are placed behind a wall).

It is well known that it is computationally expensive to calculate the whole room reverberation using only room geometry data [HSHT96] since the number of reflections grows exponentially and can reach numbers of several millions after only a few seconds. It is advantageous instead to describe room reverberation via a statistical model driven by perceptual parameters (section 5.4.8). In XML3DAUDIO, late reverberation is thus described separately in the *Room* object.

In order to simplify surface definition, only flat polygonal surfaces can be described. These are defined by a minimum of three vertex points in cartesian space defined in the *points* field of the *surface* object. The acoustical properties of surfaces are then defined in the *reflectivity* and *through* fields whether the surface acts as a reflector, an absorber or both. Surface reflectivity[12] and transmissivity[13] are described by specifying IIR filter coefficients for different incident angles. Alternatively, reflectivity and transmissivity may be defined by attenuation gains for different angle of incidence at different frequency bands, linear interpolation is then used to calculate intermediate values. From this data, a filter is then computed to apply the necessary attenuation on the sound source signal which was reflected or went through the surface, to do so several existing filter design techniques can be used [OW97].

The surface material properties can also be described in a *definition* object. This is useful when many surfaces have the same material, since the material description needs to be performed only once in the *definition* object, avoiding redundancy in the

[12]That is, the transfer function of signal attenuation due to reflection on a surface
[13]That is, the transfer function of signal attenuation after travelling through a surface

XML 3D audio scene description.

The capabilities of the *Surface* object of XML3DAUDIO are identical to that of the *AcousticMaterial* node of MPEG-4 Advanced AudioBIFS [PE02]. The semantics of the *surface* object are shown in Fig. 3.10 below.

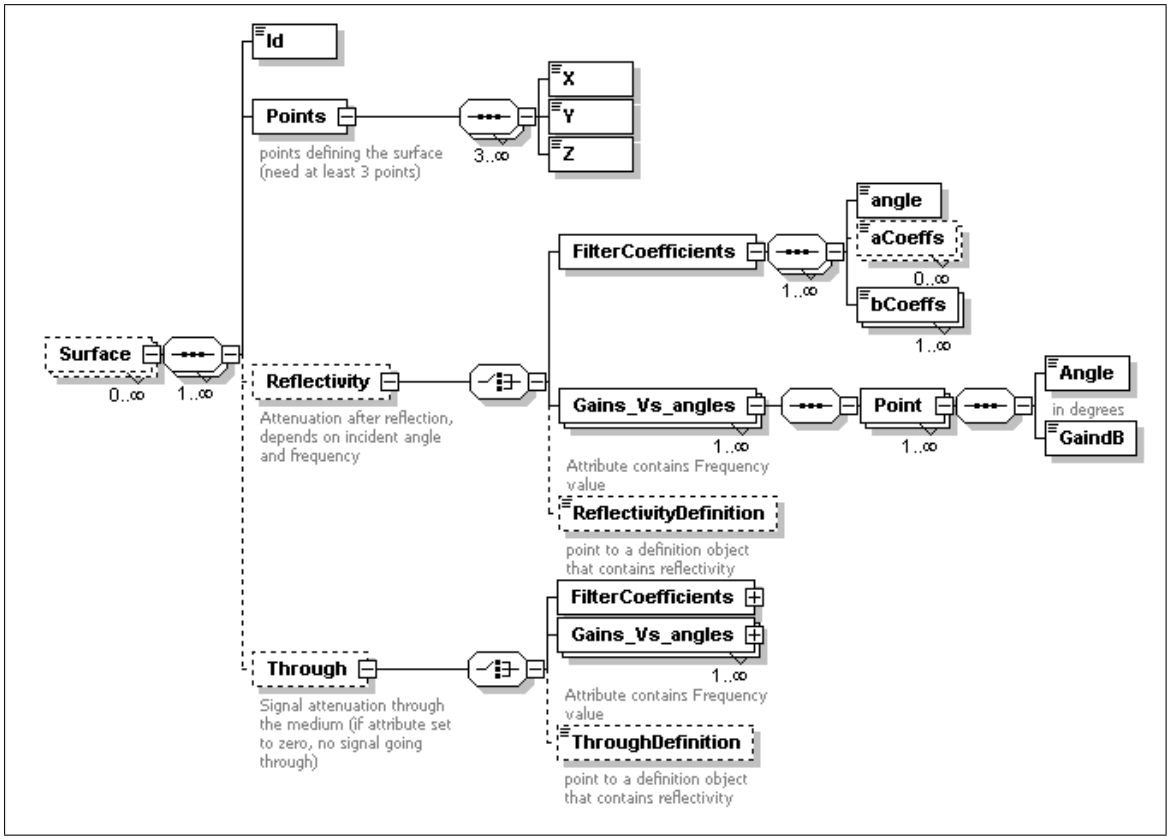


Figure 3.10: Semantics of the *Surface* object

Medium object

The *Medium* object describes properties of the virtual medium which induces attenuation and delay on the sound source signals travelling in the virtual medium of the 3D audio scene. If the *Air_Attenuation* field of the *Medium* object is set to '1', air attenuation is applied to the sound sources, otherwise air attenuation is not applied.

In the *Medium* object, the following parameters are used to describe the medium

properties: *temperature*, *humidity* and *pressure*. These parameters can be used by the scene renderer to calculate signal attenuation during propagation of source signals in the virtual medium (see section 5.4.3).

The *propagation speed* parameter controls propagation delays and has an effect on the strength of the doppler effect which is applied on moving sound sources (see section 5.4.5). The strength of the Doppler effect can also be controlled separately with the *Doppler factor* parameter. Several *Medium* objects can be defined in the scene orchestra, this is useful when describing 3D audio scenes with different acoustical environments (e.g. different rooms) which have different propagation properties due to changes in medium characteristics between rooms.

In MPEG-4 AudioBIFS, medium description is done in the *DirectiveSound* node and only includes the definition of the speed of sound and the definition of a flag that specifies if air attenuation should be applied to the source signals. XML3DAUDIO thus provides more accurate medium description since the *temperature*, *humidity* and *pressure* of the medium can be described. This, in turn can be used to describe 3D audio scenes with peculiar medium propagation properties (e.g. extremely cold environments). Medium properties may also be animated over time in the scene score to create novel special effects.

The semantics of the *medium* object are shown in Fig. 3.11.

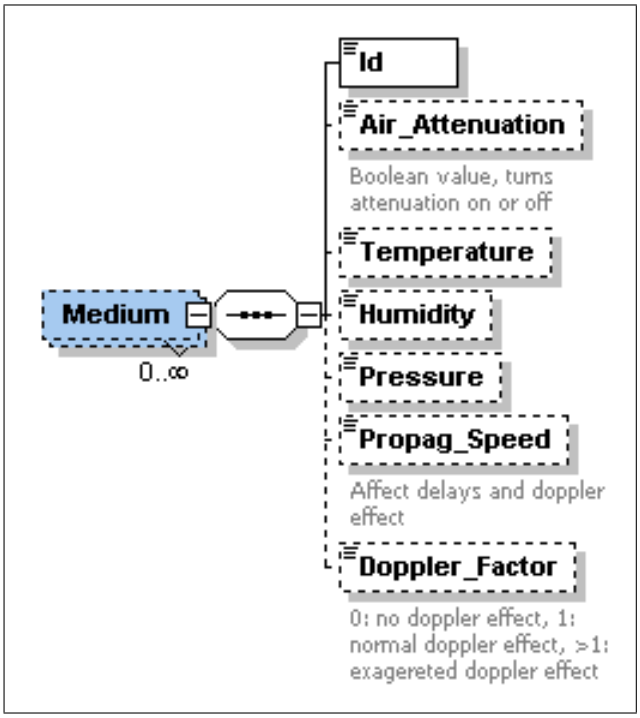


Figure 3.11: Semantics of the *Medium* object

Room object

The *Room* object is used to describe room reverberation by three means. The simplest way is to use orthogonal subjective parameters that were developed in [JW95] and which are also used in MPEG-4 advanced AudioBIFS [VH99]. These parameters are: *room presence*, *running reverberation*, *envelopment*, *late reverberance*, *heaviness* and *liveness*.

Secondly, room reverberation can be described by reverberation times (T_{60}) at different frequencies, this is the coarsest way to describe room reverberation. The last and most precise method for describing room reverberation is by means of an external B-format impulse response which contains the 3D acoustical response of a particular room. At the renderer, the source signal and the B-format impulse responses are convolved to produce the desired 3D reverberation [FT98]. The B-format

impulse responses are given as a URL pointing to an external 4-channel sound file, as this is more efficient than describing impulse responses point by point in XML syntax.

Similarly to MPEG-4 Advanced AudioBIFS [PE02], XML3DAUDIO can describe room reverberation via orthogonal perceptual parameters. However, in addition, XML3DAUDIO provides two additional methods for describing late reverberation: definition by simple reverberation times at different frequencies and a more precise way which uses B-format impulse responses. This feature can be used to precisely reproduce the measured 3D reverberation of natural environments [FT98].

The semantics of the room object are shown in Fig. 3.12 below.

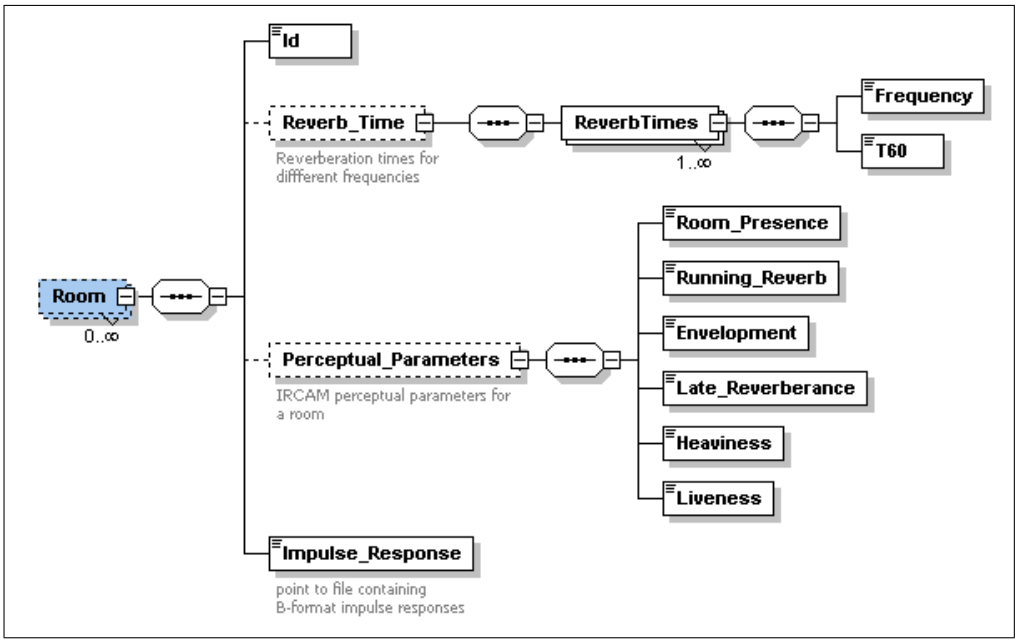


Figure 3.12: Parameters of the *Room* object

Macro-objects

In the *scene orchestra*, complex objects composed of several sound sources and acoustic surfaces may also be imported at once. These are called *macro-objects*. For example, a *macro-object* representing a car may be composed of several sound sources that represent the tyre friction, horn and exhaust pipe sounds and several surfaces defining the body of the car which in turn obstructs and reflects sounds of other sound sources; this *macro-object* example is depicted in Fig. 3.13.

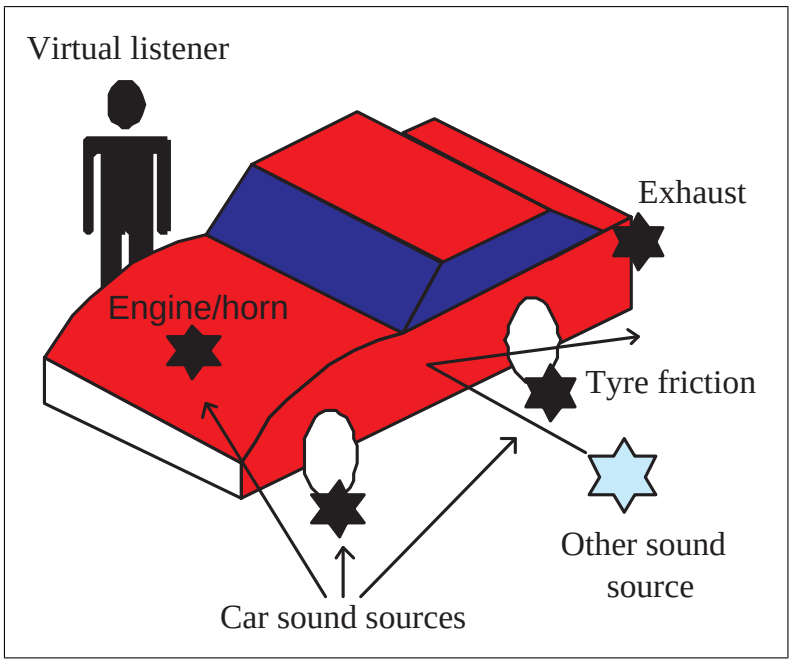


Figure 3.13: Illustration of car macro-object

Macro-objects are defined externally from a separate XML schema that is shown in Fig. 3.14. A *macro-object* is thus simply a grouping mechanism which contains several elementary objects such as sound sources, reflective surfaces, recorded scenes or even sub macro-objects.

In the 3D audio *scene orchestra*, a *macro-object* is imported via a *Macro_object* object which points to the XML description of the macro-object. Once imported in the scene, the *macro-object* is then controlled as a single object by the *scene score*,

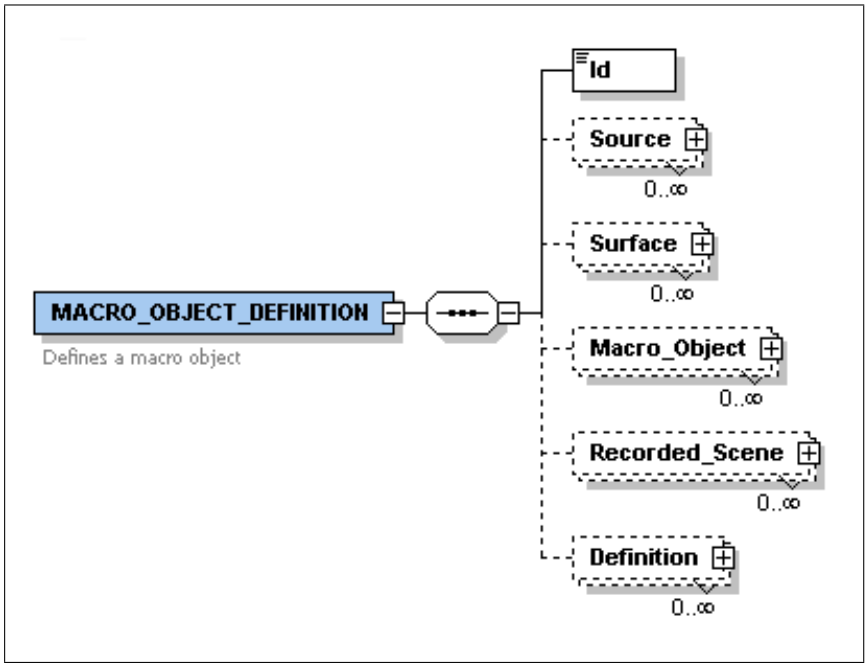


Figure 3.14: Macro-object definition schema

simplifying score descriptions. Macro-objects have *position* and *rotation* parameter fields to position the centre of the macro-object in the 3D audio scene and an *URL* field pointing to the XML description of the macro-object. Importing of macro-objects in the scene orchestra is shown in Fig. 3.15.

When defined, macro-objects contain their own sound resources, allowing complex objects to be imported at once. However, when only the template of the macro-object is wanted, it is possible to replace the resource URLs of the macro-object by other URLs pointing to different sound resources. This is achieved by specifying a list of URLs in the *SourceURLS* field. The order of definition of the URLs is important since the new URLs are attributed in the order of description of the objects in the macro-object.

A library of macro-objects (e.g. a car, a person speaking, a choir, a music orchestra etc.) has been created. This library can be used to quickly compose 3D audio scenes that contain these complex objects. The macro-object library could be further extended with more specific 3D audio objects.

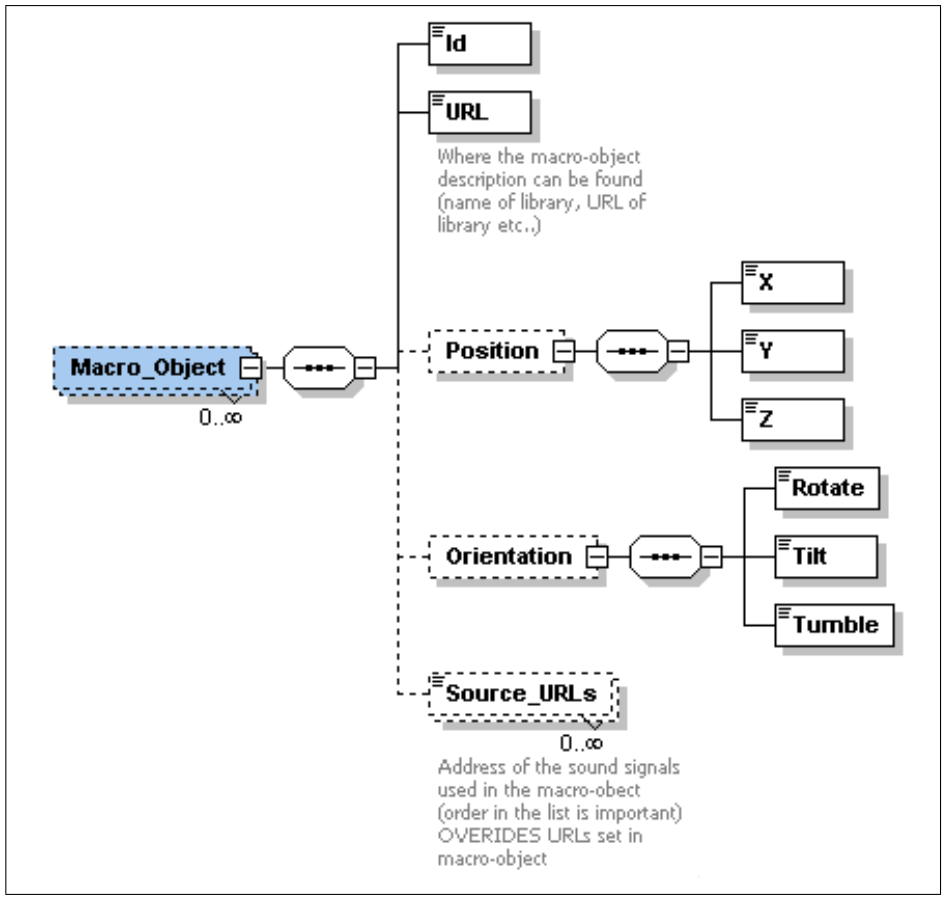


Figure 3.15: Semantics of the *macro-object* object used to import complex objects in the scene orchestra

Recorded scenes

The *Recorded_scene* object is used to incorporate sound resources that are complete 3D audio scenes in themselves. It is useful to employ the *recorded_scene* object in situations where a 3D audio scene is composed of both recorded 3D audio scenes and spatialised monaural samples (i.e. an hybrid 3D audio scene). The *Recorded_scene* object permits importing B-format[14] content of any Ambisonics order and 5.1 surround recordings.

The *URL* fields are used to point to the 3D audio scene recording which can be a local file or an online link. Optionally, the imported scene can be further rotated in three dimensions as specified in *Rotate*, *Tilt* and *Tumble* fields. These can be be used to re-orient the imported 3D audio scenes in the current 3D audio scene. Rotations operations on B-format recordings are particularly easy to perform using simple linear equations [Dan00].

The *Recorded_scene* object of XML3DAUDIO has no equivalent in MPEG-4 AudioBIFS. It is highlighted in 3.4.3 that the description of hybrid 3D audio scenes is problematic in AudioBIFS due to its channel oriented internal structure. The semantics of the *recorded scene* object are shown in Fig. 3.16.

[14] Ambisonics B-format was reviewed in section 2.3.3 and is further developed in section 5.4.1

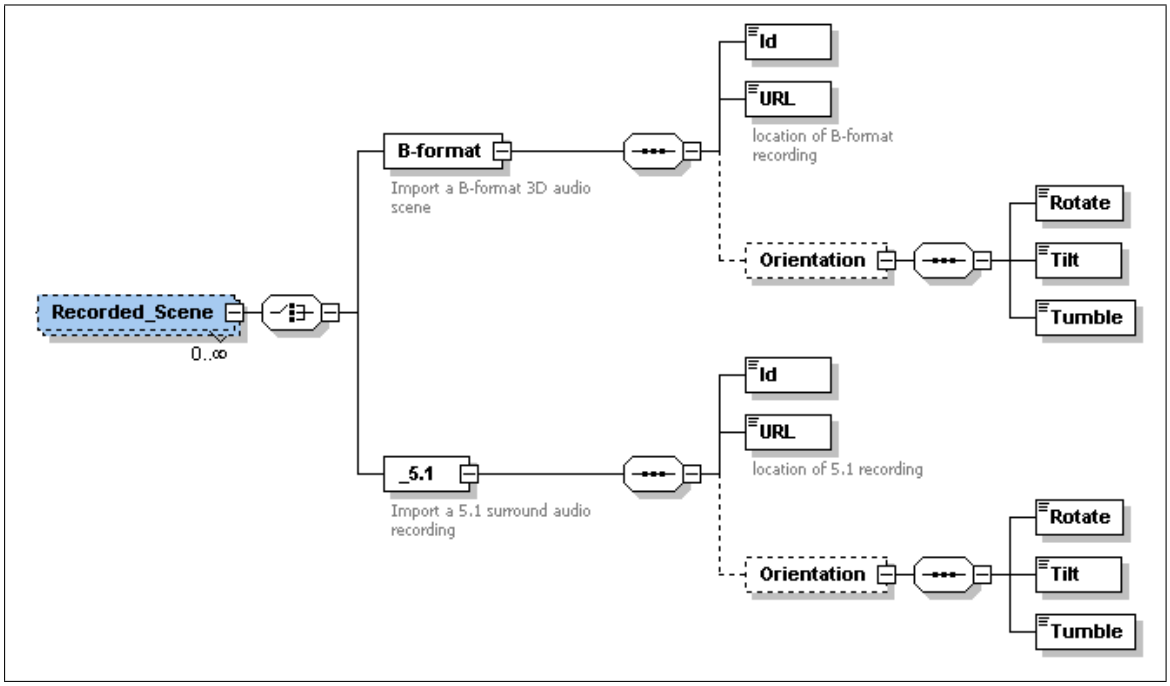


Figure 3.16: Semantics of the *recorded scene* object

Definitions

Definition objects are used to describe object properties that are frequently used by objects in the scene orchestra. For example, a number of surfaces may have the same acoustical material. This feature is used to avoid redefining the same parameters for every object. The *Definition* objects are then called by other objects by specifying the name of the *Definition* object. These are the features that can be described in the *definition* object: listener HRTF curves, sound source directivity patterns and material properties (Fig. 3.17). This feature can be compared to the PROTO mechanism [ANM97] of VRML and AudioBIFS.

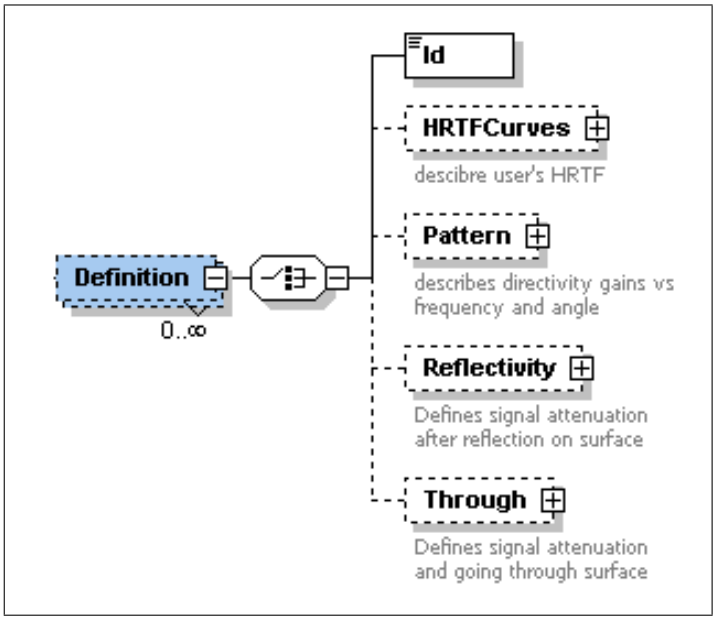


Figure 3.17: Semantics of the *definition* objects

3.3.7 3D audio scene example

The XML syntax of a simple 3D audio scene described using the new XML3DAUDIO scheme is now given. The example 3D audio scene represents the auditory scene heard by someone seating on the beach. This example scene is hybrid since it uses both a 3D audio recording and spatialised sound sources. The *scene orchestra*, which describes the scene content, contains four objects:

- A listener located at position (3,5,0) in the scene and with an head orientation facing the frontal direction (Azimuth=0, Elevation=0, Roll=0)
- An omnidirectional point sound source (*seagull*) located at position (10,5,10) at the start of the scene
- A line sound source (*beach front*) with an horizontal spatial extent of 100 m
- A B-format 3D audio recording (*beach crowd*).

The *scene score* which describes the scene timing and animation contains three lines of performance score.

- The first line of score plays the *beachfront* and *beach-crowd* sound sources at the start of the scene and for a duration of 100 seconds. In case the sound samples are shorter than this period, they are looped as specified by the *loop* parameter.
- The second line of scores plays the *seagull* sound source between 20 and 30 seconds.
- The last line of score performs the animation of the *seagull* object from its initial position [10,5,10] to position [20,10,8], in 10 seconds and starting at 20 seconds from the start of the scene.

```
<?xml version="1.0" encoding="UTF-8"?>
  <ORCHESTRA>
    <Listener>
      <Id>Guillaume</Id>
      <Position>
        <Xl>3</Xl>
        <Yl>5</Yl>
        <Zl>0</Zl>
      </Position>
      <Orientation>
        <Azimuth>0</Azimuth>
        <Elevation>0</Elevation>
        <Roll>0</Roll>
      </Orientation>
```

```
</Listener>
<Source>
    <Id>beachfront</Id>
    <URL>beachfront.wav</URL>
    <Dimensions>
        <X>0</X>
        <Y>100</Y>
        <Z>0</Z>
    </Dimensions>
</Source>
<Source>
    <Id>seagull</Id>
    <URL>http:\\dummyserver.com\seagull-stream.mp3</URL>
    <Position>
        <X>10</X>
        <Y>5</Y>
        <Z>10</Z>
    </Position>
</Source>
<Recorded_Scene>
    <B-format>
        <Id>beach-crowd</Id>
        <URL>beach-crowd.wxyz</URL>
    </B-format>
```

```
    </Recorded_Scene>
</ORCHESTRA>
<SCORE>
    <Performance_Score>
        <Line_of_Score>
            <start_time>0</start_time>
            <Duration>100</Duration>
            <Command>play</Command>
            <Object>beachfront</Object>
            <Object>beach−crowd</Object>
            <Parameter>loop</Parameter>
        </Line_of_Score>
        <Line_of_Score>
            <start_time>20</start_time>
            <Duration>30</Duration>
            <Command>play</Command>
            <Object>seagul</Object>
        </Line_of_Score>
        <Line_of_Score>
            <start_time>20</start_time>
            <Duration>30</Duration>
            <Command>move</Command>
            <Object>seagul</Object>
            <Parameter>20</Parameter>
```

```
            <Parameter>10</Parameter>
            <Parameter>8</Parameter>
          </Line_of_Score>
        </Performance_Score>
    </SCORE>
</AUDIO_SCENE>
```

3.4 Evaluation of the novel scheme

The 3D audio description capabilities of XML3DAUDIO are now compared to those of VRML and MPEG-4 Advanced AudioBIFS. Then, a 3D audio scene example is used to demonstrate the advantages of the scene orchestra and score approach over the scene graph model. Lastly, it is explained how XML3DAUDIO can describe hybrid 3D audio scenes while it is problematic to do so in MPEG-4 AudioBIFS.

3.4.1 Feature comparison with VRML and MPEG-4

The 3D audio description capabilities of XML3DAUDIO were detailed in the description of the objects that can be used in the scene orchestra (section 3.3.6). The description capabilities of the scheme are summarised and compared against the 3D audio description capabilities of the VRML and MPEG-4 AudioBIFS standards in Fig. 3.18. It can first be seen that the 3D audio description capabilities of VRML are very limited. Secondly, it can be seen that XML3DAUDIO includes all the advanced features of MPEG-4 AudioBIFS while providing new features such as: the ability to describe sound source extent, the ability to import other 3D audio content in the current scene, the ability to construct 3D audio scenes algorithmically, the ability to describe the listener HRTFs and the ability to describe the medium in a more precise manner. While providing these extra features, it is shown in section 3.4.2

that XML3DAUDIO has a much simpler declarative syntax compared to VRML and MPEG-4 BIFS because it is not based on the scene graph model.

Description Feature	VRML	MPEG-4 AudioBIFS	XML3DAUDIO
Sound source directivity	Simple (Ellipsoidal model)	Full (Frequency dependent)	Full (Frequency dependent)
Sound source extent	X	Soon (in Version 3)	3D box definition
Acoustic material	X	Full: Frequency and angle of incidence dependent	Full: Frequency and angle of incidence varying
Medium	X	Speed of sound	Full: speed of sound, temperature, humidity, pressure
Acoustical environments	X	With a 3D box or a list of surface objects	With a 3D box or a list of surface objects
Reverberation	X	Perceptual parameters	Perceptual parameters, reverberation times, B-format impulse responses
3D audio scene import	X	X	Import B-format or 5.1 sound scene in current scene
Algorithmic scene composition	X	X	From the scene score, using scene composition opcodes
Listener's HRTF	X	X	In the *Listener* object

Figure 3.18: Comparison of 3D audio scene description capabilities between VRML, MPEG-4 AudioBIFS and the novel scheme

3.4.2 Simplification of scene description by the novel scheme

To illustrate the possible simplification of scene descriptions using XML3DAUDIO, a simple 3D audio scene example containing three sound sources is now used as an example. Firstly, the example scene is described with the scene orchestra and score approach, then with the scene graph approach. Comparisons in terms of complexity and re-usability between the two scene description approaches are then made.

3D audio scene example

The example scene is a simple 3D audio scene which has a total duration of 20 s and which contains three omnidirectional sound sources. During the course of the scene, each of the three sound source is played twice and is moved once; from the default position (0,0,0) to a new position (x,y,z). The signals emitted by the sound sources originate from monaural sound files called *source1.wav*, *source2.wav* and *source3.wav* which are stored on a server and are accessible by specifying URLs. The animation and sequencing of the 3D audio scene example are depicted in Fig. 3.19.

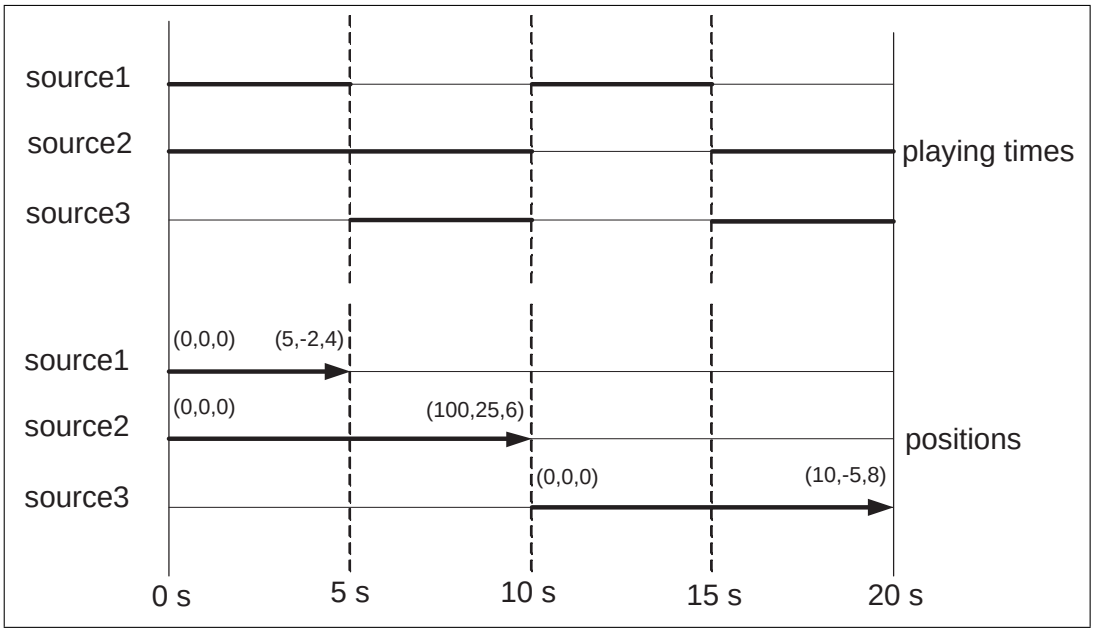


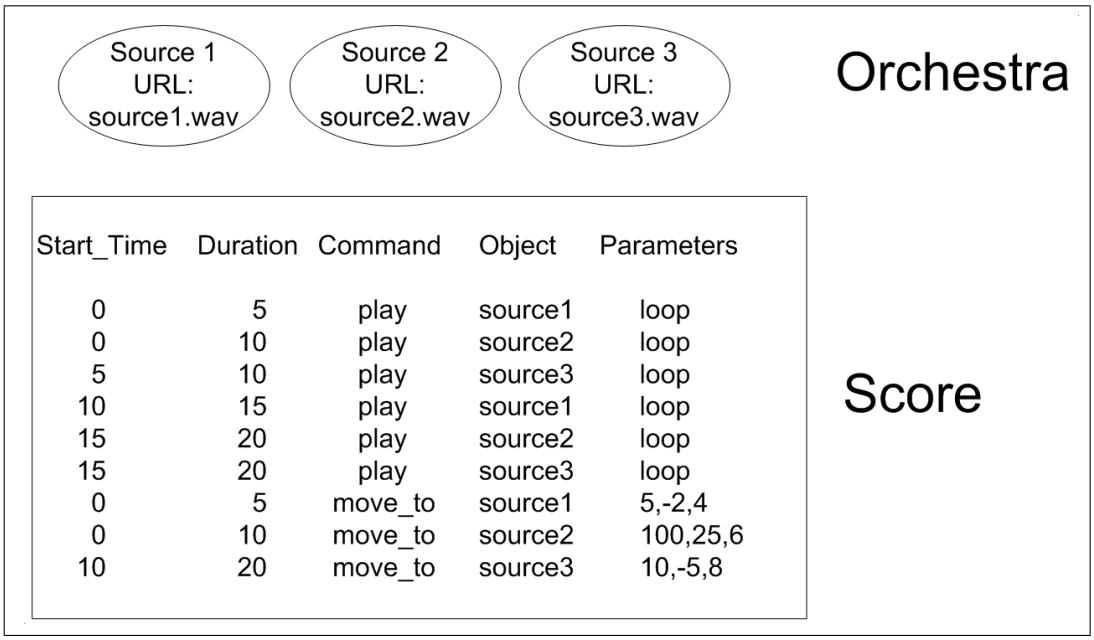
Figure 3.19: Playing times and animation of the example 3D audio scene

Description of the example scene using the scene orchestra and score approach

Using XML3DAUDIO, the example 3D audio scene is described by three *source* objects[15] defined in the *scene orchestra*.

[15]The semantics of the source object were described in section 3.3.6

The *scene score* then contains six lines of score defining the playing times of the sound sources and three lines of score describing sound source trajectories. The orchestra and score of the example scene is depicted in Fig. 3.20.



Start_Time	Duration	Command	Object	Parameters
0	5	play	source1	loop
0	10	play	source2	loop
5	10	play	source3	loop
10	15	play	source1	loop
15	20	play	source2	loop
15	20	play	source3	loop
0	5	move_to	source1	5,-2,4
0	10	move_to	source2	100,25,6
10	20	move_to	source3	10,-5,8

Figure 3.20: Scene orchestra and score description of the 3D audio scene example

Description of the example scene using the scene graph approach

The same 3D audio scene example is now described using the scene graph approach. The diagram of the resulting scene graph is depicted in Fig. 3.21.

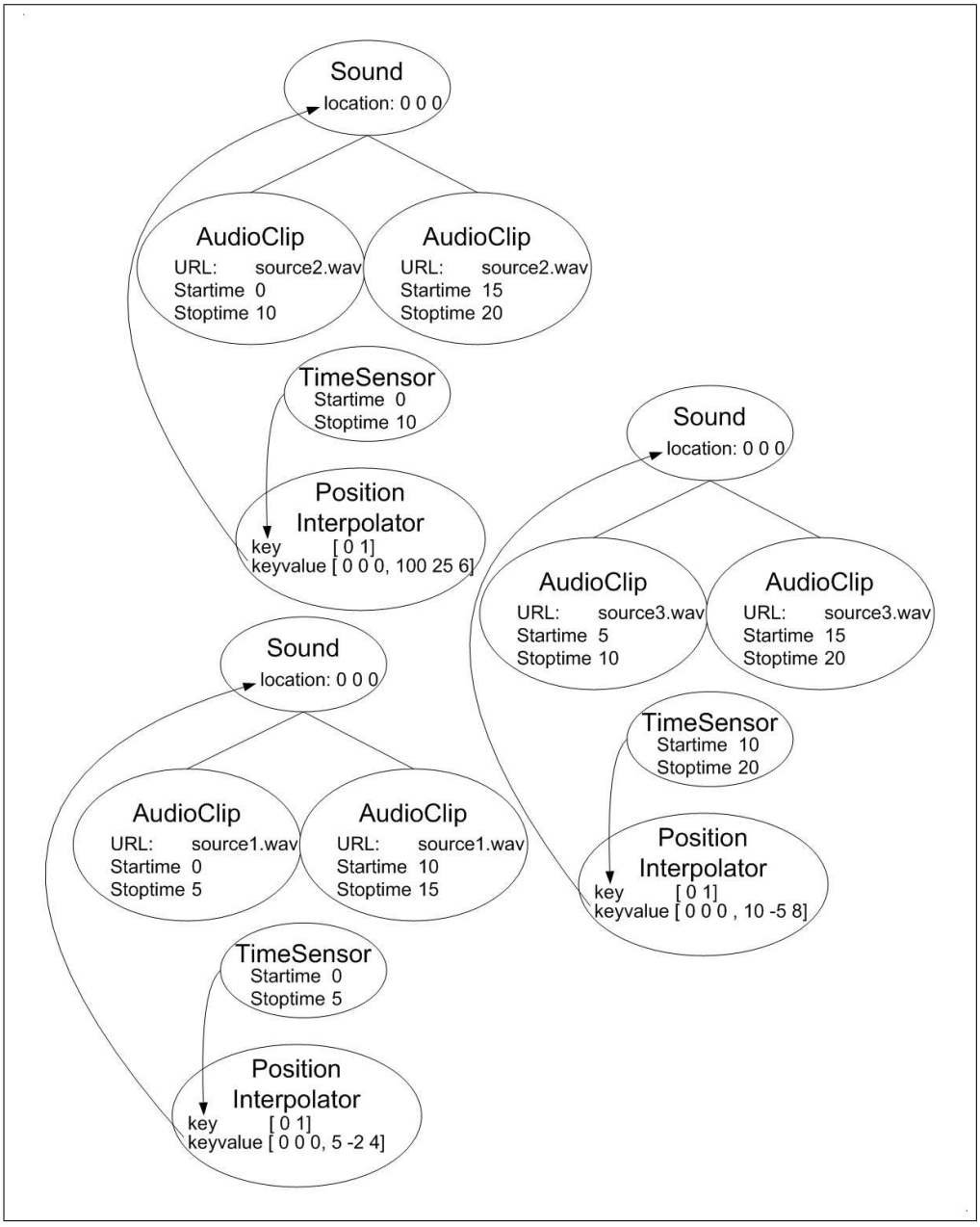


Figure 3.21: Scene graph description of the 3D audio scene example

To describe the example scene using the scene graph approach, two *AudioClip* nodes are needed per *source* object to address the fact that each sound source is played twice. The fields of the *AudioClip* node contain start and stop times defining the sequencing of the sounds to be played. The animation of one sound source is then achieved with a *TimeSensor* node that produces a clock signal during a period defined in the start and stop time fields of the object. The *TimeSensor* clock events[16] are then routed to a *PositionInterpolator* node. The latter outputs interpolated coordinate value between the start and target positions. The interpolated position values are finally routed to the *position* field of the *sound* node to produce the displacement of the sound source.

Comparison of the two approaches

At first sight, it can be seen that scene animation description using the scene graph approach is more complex that with the new scene orchestra and score approach.

To describe the sequencing of playing times of sound sources, a new *AudioClip* node was required each time a sound source was played, resulting in description redundancy. A solution to this problem would be the use of a single *AudioClip* object and several *timeSensor* and interpolator nodes changing the start and stop fields of the *AudioClip* objects at certain times in the scene in order to perform sequencing of the sound source. It can be seen however, that this is a complex and non-viable solution for describing intricate sequencing of sound sources.

Then, to describe the trajectories of the sound sources of the example scene, one *timeSensor* node, one interpolator node and two routing mechanisms were required per sound source. This again is complex is comparison to the scene orchestra and score approach where a single line of score was required. The scene graph approach can thus be seen as being inefficient in describing the temporal behaviour of scenes.

[16] these are called fraction_changed events [ANM97]

In the scene orchestra and score approach, it can be seen that increasing the complexity of the scene temporal behaviour simply requires more lines of scene score. Besides, this increase in temporal complexity is not reflected in the semantic structure of the scene description such as in the scene graph approach. Scene description is thus more efficient with the scene orchestra and score approach than with the scene graph approach. This, in turn allows the scene renderer to perform faster parsing of the scene and less complicated scene parsing tasks are required.

Another important observation is that, in the scene graph description of the example 3D audio scene (Fig. 3.21), it can be seen that the temporal information of the scene is spread out in the various fields of the *AudioClip*, *TimeSensor* and *Interpolator* objects. This temporal information can be deeply nested in sub-branches of the scene graph. This again requires more complex tasks at the scene renderer to parse the scene description.

For a scene author who wants to modify the behaviour of scene animation, the scene graph description must first be reverse engineered to understand its temporal behaviour and the intentions of the original scene author. This might prove to be a difficult and time consuming task if animation is performed using many *TimeSensor*, Interpolation nodes and routes which interact in complex ways.

In contrast, in the scene orchestra and score approach (Fig. 3.21), the scene temporal information is clearly separated from the scene content and structure information and is neatly centralised in the scene score. Thus the scene timing may be easily understood and re-authored; this can be done by hand, directly in the XML scene score description.

The issues that have been highlighted regarding scene animation in the scene graph model were also discovered by the designers of the MPEG-4 standard [WBS02]. Thus, MPEG-4 BIFS was equipped with two additional scene animation mechanisms known as BIFS-Anim and BIFS-Commands [MPE99, SVH99] (see 2.4.2). These mechanisms

can be used to simplify scene animation description. For example, BIFS-Commands can be issued to modify the start and stop times of *AudioClip* nodes so that sound sources can be sequenced. Similarly, BIFS-Anim streams can be used to control the position of the sound sources as described in Fig. 3.19. These animation mechanisms are somehow similar to the scene score modifying the content of the scene (i.e. orchestra). However, it can first be seen that the sequencing and animation descriptions of sound sources are separated in BIFS-Commands and BIFS-Anim respectively. This contrasts with XML3DAUDIO, where all temporal information is clearly centralised in the scene score, permitting easy re-authoring of the scene temporal behaviour.

Secondly, BIFS-Commands and BIFS-Anim mechanisms require the MPEG-4 system layers (section 2.4.2) to be usable. For example, a BIFS-Command does not explicitly contains the time at which the action should be executed but must be associated with a time stamp of the binary stream [PE02]. Thus scene animation using BIFS-Anim and BIFS-Commands is still decentralised and requires a complex framework, this contrasts with XML3DAUDIO where scene animation is centralised and is simply described in XML.

Summary

Using a simple animated 3D audio scene example, the complexity of the scene graph approach was compared against the novel scene orchestra and score approach. It was shown that scene animation performed with *TimeSensor* and Interpolator nodes creates complex and tangled scene graphs and that the temporal information of the scene is spread out in the whole scene graph. Therefore, scene animation that is performed intrinsically using animation nodes is clearly not a viable solution for describing complex scene temporal behaviours. In VRML and MPEG-4 BIFS, a partial solution to this problem would be the use of a *Script* nodes that contains temporal information and route events to scene objects at certain times. It can be seen that this may be done in arbitrary and different ways by several scene authors; and thus it is not a standardised and universal solution. In addition, the use of *Script* nodes

does not address the issue that the scene temporal data is decentralised.

Being aware of these issues, the creators of the MPEG-4 standard added two additional scene animation mechanisms: BIFS-Anim and BIFS-commands. It has been shown that although greatly simplifying the scene graph, these mechanisms are unique to MPEG-4 and are heavily dependant on the MPEG-4 system layers (section 2.4.2). Besides, the scene temporal information is still not described in a centralised way.

In comparison, XML3DAUDIO describes the temporal behaviour of scenes without relying on external mechanisms. By providing a centralised data storage space describing the sequencing and the animation of 3D audio scenes, the proposed scheme permits a simpler description of the 3D audio scene temporal behaviour. This in turn, improves the efficiency of the scene renderer and allows scene animation to be easily re-authored. Since the novel scheme is based on XML, scene animation can be comfortably modified with a simple text editor. This contrasts with the complex binary framework of the MPEG-4 standard which inevitably requires special authoring tools.

3.4.3 Description of hybrid 3D audio scenes

This section identifies a problem inherent to the MPEG-4 AudioBIFS standard when describing hybrid 3D audio scenes and shows how the novel scheme avoids this issue.

Hybrid 3D audio scenes are the combination of 3D audio recordings (e.g. B-format[17]) with spatialised monaural sound sources. This can be used, for instance, to add synthetic sound sources to a natural 3D audio recording that has been captured by a Soundfield microphone (see 2.3.3).

In XML3DAUDIO, hybrid scenes are easily described by using the *Recorded_Scene* object of the scene orchestra which is used to import 3D audio recordings in the current scene. After importing a 3D audio recording in the scene orchestra, it is

[17]See section 2.3.3

treated as a normal scene object and can be further manipulated (e.g. rotated) and animated by the scene score. The semantics of the *Recorded_Scene* object were described in section 3.3.6 and its use was also shown in an hybrid 3D audio scene example described in section 3.3.7.

To render the hybrid 3D audio scene from its description, the scene renderer then decodes the 3D audio recording and spatialises sound sources for the target terminal configuration; this is illustrated in Fig. 3.22.

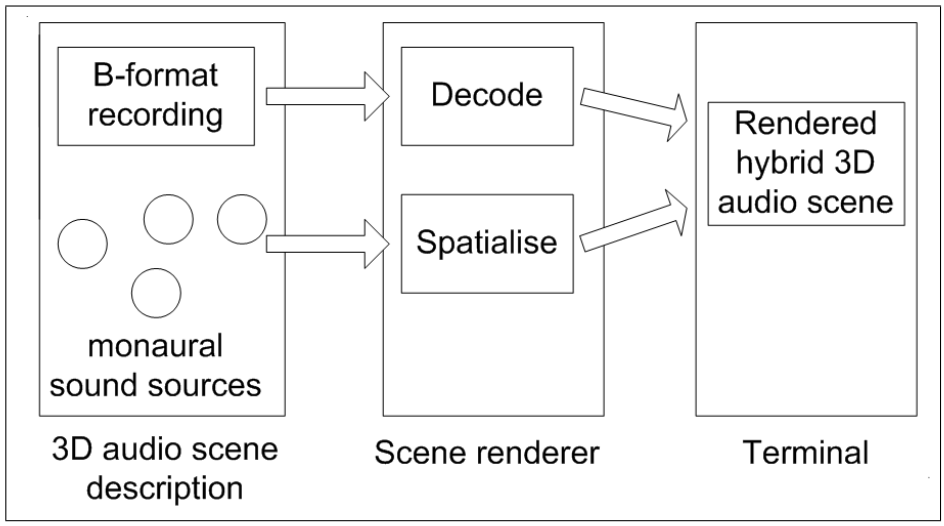


Figure 3.22: Illustration of the hybrid 3D audio scene rendering process

In MPEG-4 AudioBIFS however, there exists no scene object that allows importing of 3D audio recordings which can then be treated as scene objects. Thus, in order to devise hybrid 3D audio scenes in AudioBIFS, 3D audio recordings have to be treated and imported as sets of audio channels (for instance using four AudioClip nodes to import a first order B-format recording). This is problematic since in order to later decode the imported 3D audio recording (Fig.3.22) represented by several audio channels to the target speaker configuration, details of the decoding process must be defined in the AudioBIFS scene itself. For instance, B-format could be decoded in AudioBIFS using an AudioFX node or several AudioMix nodes to perform Ambisonics de-matrixing. As a consequence, several AudioBIFS scene descriptions

are required for different target speaker configurations; this defeats the purpose of encoding 3D audio scenes in an object oriented way in the first place.

In XML3DAUDIO, 3D audio recordings are treated as objects and not as sets of audio channels. This higher level of abstraction in turn allows the same scene description to be decoded by different scene renderers since details of the 3D audio recording channels and of the decoding process are not present in the scene description.

In conclusion, the description of hybrid 3D audio scenes in AudioBIFS is not advisable without the a-priori knowledge of the terminal configuration. In contrast, XML3DAUDIO can describe hybrid 3D audio scenes independently of the terminal configuration.

3.5 Use of the proposed scheme as a meta-data annotation scheme for 3D audio content

3.5.1 Introduction

The aims and format of the novel 3D audio scheme have been described. While the new scheme allows full description of 3D audio scenes in an object-oriented way for rendering purposes (chapter 5), an alternate use of the scheme as a 3D audio content meta-data scheme is now detailed. This proposed use of the scheme was published in [PB04c].

While the quantity of 3D and surround[18] audio material is rapidly growing, there is still no established method for describing the spatial content, acoustical properties and the temporal structure of 3D audio scenes. Generation of meta data for 3D

[18] i.e. 5.1 surround movie sound tracks

audio would, however, allow automatic classification and content retrieval of 3D audio content.

The scene graph model (see section 2.4.1) is unfit for annotating 3D audio content since several scene graph descriptions may lead to the same 3D audio scene (see 2.4.1), furthermore, the temporal data of the scene can be deeply nested within fields of objects (section 3.4.2), increasing the time and cost to access this data. The scene orchestra and score approach[19] in contrast, provides a centralised and chronological description of the scene temporal data. This data is separated from the scene orchestra which describes the scene content. This allows the scene structural and temporal data to be searched and processed separately by content retrieval algorithms, improving the efficiency of the 3D audio meta-data exploitation.

Three scenarios where 3D audio meta-data can be generated are now identified: firstly, meta-data may be generated from an existing multi-channel 3D audio recording. Secondly, meta-data may be generated from an object-oriented scene description, such as a MPEG-4 AudioBIFS 3D audio scene. Lastly, meta-data may be produced during production of the 3D audio content. The three scenarios are now outlined.

Meta-data generation at post production

In the case where meta-data needs to be generated from an existing 3D audio recording or surround sound track (e.g. a 5.1 movie track) that has already been composed or recorded, it is necessary to extract the meta-data information from the 3D audio recording itself. This can prove to be a highly complex task and human intervention is often still required. However, a certain degree of automation can be achieved by the use of special digital signal processing techniques. For example, with B-format recordings, it is possible to derive virtual microphone signals that allow 'zooming' on a particular area of the recorded sound field [McG02]. With the help of adaptive techniques, this approach could be used to track and follow the positions of sound

[19]Presented in section 3.3

sources in the sound field and to generate meta-data automatically.

Meta-data generation from object-oriented 3D audio scene descriptions

In the case of a 3D audio scene described in VRML or MPEG-4, a solution to meta-data generation is to process the scene description by a meta-data engine, this is shown in Fig. 3.23. MPEG-4 XMT [KWC00], the textual XML format of MPEG-4 BIFS could, for instance, be processed by using XSLT stylesheets[20] [Tid01] to produce 3D audio meta-data which conforms to the XML schema of the novel 3D audio scene description scheme.

Meta-data generation during content creation

The 3D audio meta-data can also be generated at the mixing or production stage along with the produced 3D audio material. This approach produces the most accurate meta-data since the parameters of the 3D audio scene are precisely known. This technique is illustrated in Fig. 3.24; it is shown that a special plug-in in the scene authoring software automatically generates XML meta-data about the authored scene. This approach has the advantage that human intervention is minimal and that the generated meta-data is more precise than if extracted from the 3D audio content.

[20] An XSLT stylesheet is an XML technique which defines rules to transform an XML document into another

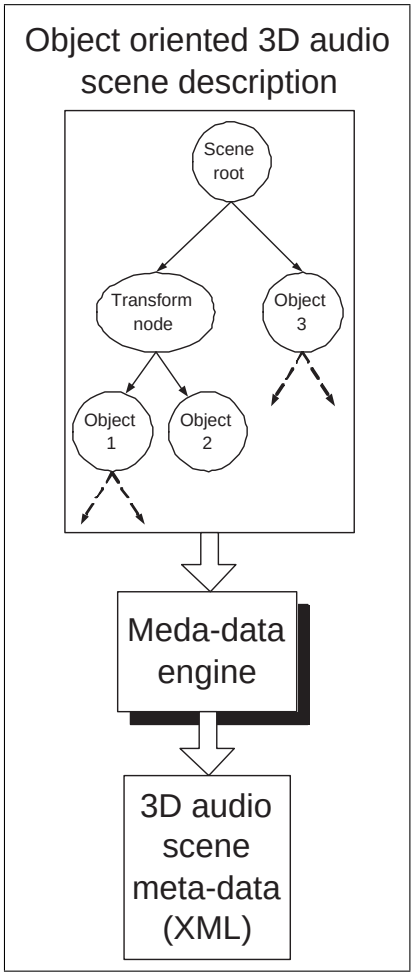


Figure 3.23: Illustration of XML meta-data generation from the 3D audio scene description

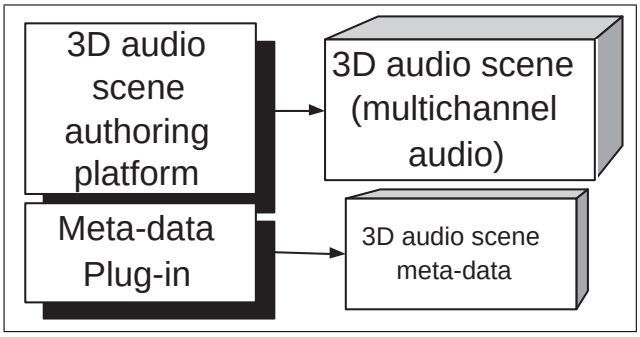


Figure 3.24: Generation of 3D audio meta-data at the authoring stage

3.6 Summary

In this chapter a novel XML based scheme for describing time varying 3D audio scenes was presented. It was shown that the new scheme does not follow a traditional scene graph model found in VRML or MPEG-4 AudioBIFS but a new scene orchestra and score approach which is inspired from the sound synthesis language CSound. It was highlighted that, in contrast to the scene graph model, the scene orchestra and score model separates the scene content description from the scene temporal description. It was then shown that this simplifies the syntactic structure of the scene description and allows scenes to be easily re-authored. It was also shown in several 3D audio scene examples that the scene graph model, which was initially designed for describing interactive 3D graphical scenes, is inefficient and cumbersome when applied to the description of animated 3D audio scenes.

In XML3DAUDIO, the scene score can be used to compose 3D audio scenes algorithmically using special composition commands. This provides a powerful mean to describe complex 3D audio scenes with a simple description. This novel technique then allows the scheme to be later extended with further scene score commands without changing its basic structure. It can be noted that the scene orchestra and score approach could be applied not only to 3D audio scenes and could be employed to describe any time varying structure which has intrinsic hierarchical relationships (e.g. animated visual scenes).

The 3D audio description capabilities of XML3DAUDIO were described and it was shown that they are superior to that of MPEG-4 Advanced AudioBIFS.

A practical implementation of XML3DAUDIO and the digital signal processes required to render 3D audio scenes are detailed in chapter 5.

Chapter 4

Perception of sound source extent and shape

4.1 Introduction

The psychoacoustic concepts involved in the perception of sound source extent were reviewed in section 2.6. It was shown that, while the perception of the size of a single sound source was linked to a non spatial cue called tonal volume, the global extent of multiple uncorrelated sound sources could, under certain conditions, merge into a single sound source which extent was defined by the positions of the sound sources. From these observations, a technique was proposed (reviewed in section 2.12) to control the extent of sound sources in 3D audio displays. This technique, to produce broad sound sources, relies on decorrelating a monaural source signal into several perceptually equal signal replicas which are then spatialised at different positions. It was shown in section 2.12 that placing the decorrelated sound sources in 1, 2, and 3D arrays, multidimensional source extents could potentially be obtained. The author goes further and proposes a new hypothesis which states that, by arranging the positions of the decorrelated sound sources into particular patterns, sound sources with certain apparent *shapes* are obtained. Until now, a very limited literature (which was reviewed in section 2.8.6) attempted to study the apparent shape of sound sources, and results were inconclusive. By using the decorrelated point source technique, it

is expected that better stimuli can be devised compared to this initial research and thus, sound source shape perception is allowed to be studied appropriately.

This chapter presents new psychoacoustic experiments where the perception of one-dimensional and two-dimensional apparent sound source extent (including shapes) is studied. The main question that the first experiments try to answer is "Is there a substantial shift between the intended sound source extent and the extent actually perceived by subjects ?", or in other words, "Are subjects able to localise the position of the decorrelated sound sources with a high enough precision so that an accurate apparent source extent or shape is perceived ?". To answer these questions, the experiments are first carried out in a best case scenario, that is, the decorrelated sound sources are implemented on distinct speakers placed in front[1] of subjects so as to maximise sound localisation ability by subjects. This procedure allows focusing on the perception of apparent source extent alone and avoids errors introduced by spatialisation. In a second time, since the rendering of source extent and shape is targeted to be used in 3D audio displays, the experiments are repeated in a more realistic scenario; that is, virtual (i.e. spatialised) sound sources are used to produce stimuli which have apparent extents and shapes, and which furthermore, are located on the sides, back, and above subjects.

The research described in experiments detailed in sections 4.3, 4.5, 4.7 and 4.8 was originally carried out for the MPEG standardisation body to study the need and feasibility of implementing sound source extent capabilities in MPEG-4 AudioBIFS [PB03, PS03, PS02, PSS02, PSS03]. The outcome of this research is the creation of a new AudioBIFS node called *WideSound* (in version 3 of AudioBIFS); details of this research outcome are detailed in section 4.10. The experiments for MPEG were solely designed by the author, however, the author acknowledges the kind participation of Jens Spille of Thomson Multimedia (Germany) and Jeongil Seo of ETRI (Korea) for repeating the experiments at their respective laboratories.

[1] where localisation accuracy is best

4.2 Overview of the experiments

The novel experiments studying the perception and rendering of apparent sound source extent and shape in 3D audio displays are now outlined.

- Experiment 1: **Perception of one-dimensional horizontal sound source extent** (section 4.3) The first experiment studies the precision at which subjects perceive the horizontal extent of sound sources. Spatially extended stimuli are produced on a 7-speaker horizontal array with a density of speakers so as to maximise sound localisation ability by subjects. Effects of decorrelated sound source density, signal type and signal loudness are also studied. These different stimuli aim at studying the robustness of the decorrelated point source technique when using different source signals.

- Experiment 2: **perception of horizontal, vertical and 2D sound source extent** (section 4.4) This experiment studies the perception of sound sources with horizontal, vertical and rectangular apparent extents. The experiment is performed in a real 3D audio rendering system scenario, that is, *spatialised* decorrelated sound sources are used to create the stimuli. Extended sound sources are presented at the front, back, sides and above subjects so as to study the effects of sound localisation accuracy on the perceived extent of sound sources. In the conclusion of the experiment, it is suggested that rendering apparent sound source extent in 3D audio displays could be used for data sonification purposes.

- Experiment 3: **perception of sound source shape using real decorrelated sound sources** (section 4.5) This experiment studies the ability of subjects to identify sound sources having apparent shapes. The sound source shape stimuli are created using real sound sources (i.e. speakers) so as to maximise sound localisation ability by subjects. The experiment is repeated for four types of signal (low passed noise, high passed noise, broadband noise and a blues guitar

recording) so as to study the effects of signal type on apparent source shape perception. The experiment is then repeated with stimuli presented at the front and back of subjects so as to study the effects of sound localisation accuracy on apparent source shape perception.

- Experiment 4: **perception of sound source shape using virtual decorrelated sound sources** (section 4.6) This experiment studies the ability of subjects to identify sound sources having apparent shapes. Unlike experiment 3, however, virtual (i.e. spatialised) decorrelated sound sources are used to create the sound source shape stimuli. This is in order to study the perception of apparent sound source shape in the context of real 3D auditory displays, and indeed, to study whether the decorrelated point source method can be used in conjunction with spatialisation. Effects of inter-source correlation are also studied by repeating the experiment for decorrelated and correlated sound sources.

- Experiment 5: **improvement in 3D audio scene realism by using extended sound sources** (section 4.7) This experiment studies the perceptual benefits of using extended sound sources in 3D audio scenes. To do so, subjects are asked to compare in terms of naturalness 3D audio scenes that used spatially extended sound sources and 3D audio scenes that used only point sound sources.

- Experiment 6: **perceptual effects of dynamic decorrelation** (section 4.8) This experiment studies the perceptual effects of dynamic decorrelation. Dynamic decorrelation was reviewed in section 2.13.6. Subjects are asked to judge the naturalness and fatigue of audio scenes that use dynamic decorrelation, fixed decorrelation and no decorrelation. This experiment aims at studying the usefulness of using dynamic decorrelation when rendering sound sources with apparent extents and shapes.

- Experiment 7: **perceptual effects of time-varying decorrelation** (section

4.9) This last experiment studies the effects of time varying decorrelation, that is, decorrelation which strength varies over time. Time varying decorrelation was reviewed in section 2.13.8. The experiment allowed finding the time constant at which the binaural system is able to perceive changes in the inter-aural cross-correlation coefficient (IACC). This finding has applications in designing decorrelation filters and in acoustical engineering.

The new sound source extent description capabilities that were added to MPEG-4 AudioBIFS as a result of this work are detailed in section 4.10.

4.3 Experiment 1: Perception of one-dimensional horizontal sound source extent

4.3.1 Aims

The aim of this experiment is to assess the precision by which artificially produced horizontal sound source extent can be rendered in 3D auditory displays. To do so, the *perceived* extent of artificially produced broad sound sources is studied. The experiment is repeated for different point source densities, signal types, and signal loudness so as to study the effects of these parameters on the perception of artificially rendered sound source extent.

4.3.2 Apparatus

To perform this experiment, a seven speaker horizontal array was used (Fig. 4.1). Speakers were spaced with a 30-degree angle and equidistantly placed at 1.4 m from the subject's head. Subjects were placed at the centre of the speaker array and facing the central speaker. High quality monitoring speakers (Genelec 1029A) equalised

for loudness were used. The rooms where the experiment took place (University of Wollongong and Thomson in Germany) were not anechoic but were soundproofed.

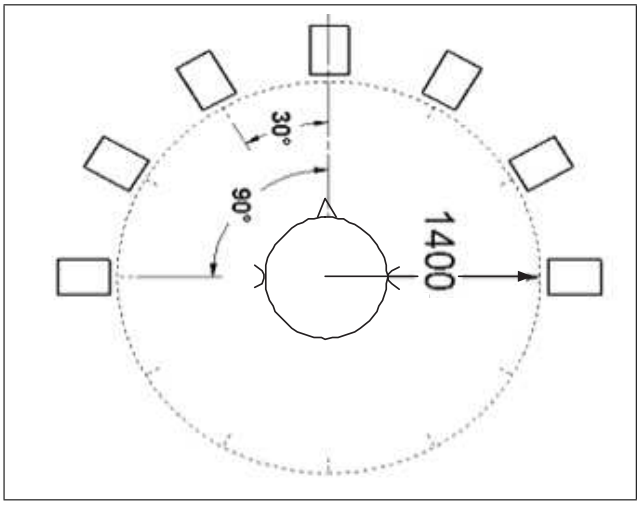


Figure 4.1: Seven-speaker horizontal array apparatus used in the experiment studying the perception of horizontal sound source extent

4.3.3 Stimuli

The decorrelated point source technique (reviewed in 2.12) was used to produce sound sources with horizontal extents of 0, 10, 30, 60, 90, 120, 150 and 180 degrees with a variable number of decorrelated point sources. A decorrelation filterbank based on time invariant all-pass FIR filters (see 2.13.2) was used to produce decorrelated signals. The filters had 256 taps, and a 44.1kHz sampling frequency was used. To place the decorrelated point sources on the speaker array, a regular constant power stereo panning scheme [Wes98] was used between pairs of speakers. Three point source densities were employed: 3 point sources (regardless of the extent), one point source per 30 degrees and one point source per 10 degrees (highest density). This resulted in 21 types of horizontally extended stimuli being created; these are depicted in Fig. 4.2. The obtained broad sound sources were equalised for loudness so that the number of point sound sources used to create the stimuli did not affect their total

loudness. Two types of signal (white noise and the recording of a large crowd) played at two levels (0dB and -6dB attenuation) were used to create the stimuli. In all, 84 different stimuli were thus presented to the subjects. The duration of each created stimulus was 10 seconds.

4.3.4 Procedure

All 84 sequences were played in random order. For each sequence, subjects were asked to assess source extent in an absolute way, that is, they were asked to draw the extent of each presented stimuli. This was judged to be the most adequate elicitation method since subjects had to simply draw the source extents that were perceived and did not have to think in terms of angles and degrees; this would have likely been inaccurate and subject to personal variations. Using this technique, complex cases when sound sources were not perceived as one (i.e. with a gap) could also be transcribed. The answer sheet for drawing results is shown in Fig. 4.3. Head rotations were allowed, and no visual masking was used. A particular sequence could be repeated on demand of the subjects. Twenty-three subjects with normal hearing and various knowledge backgrounds and ages (ten subjects at the University of Wollongong and thirteen at Thomson) took part in the experiment.

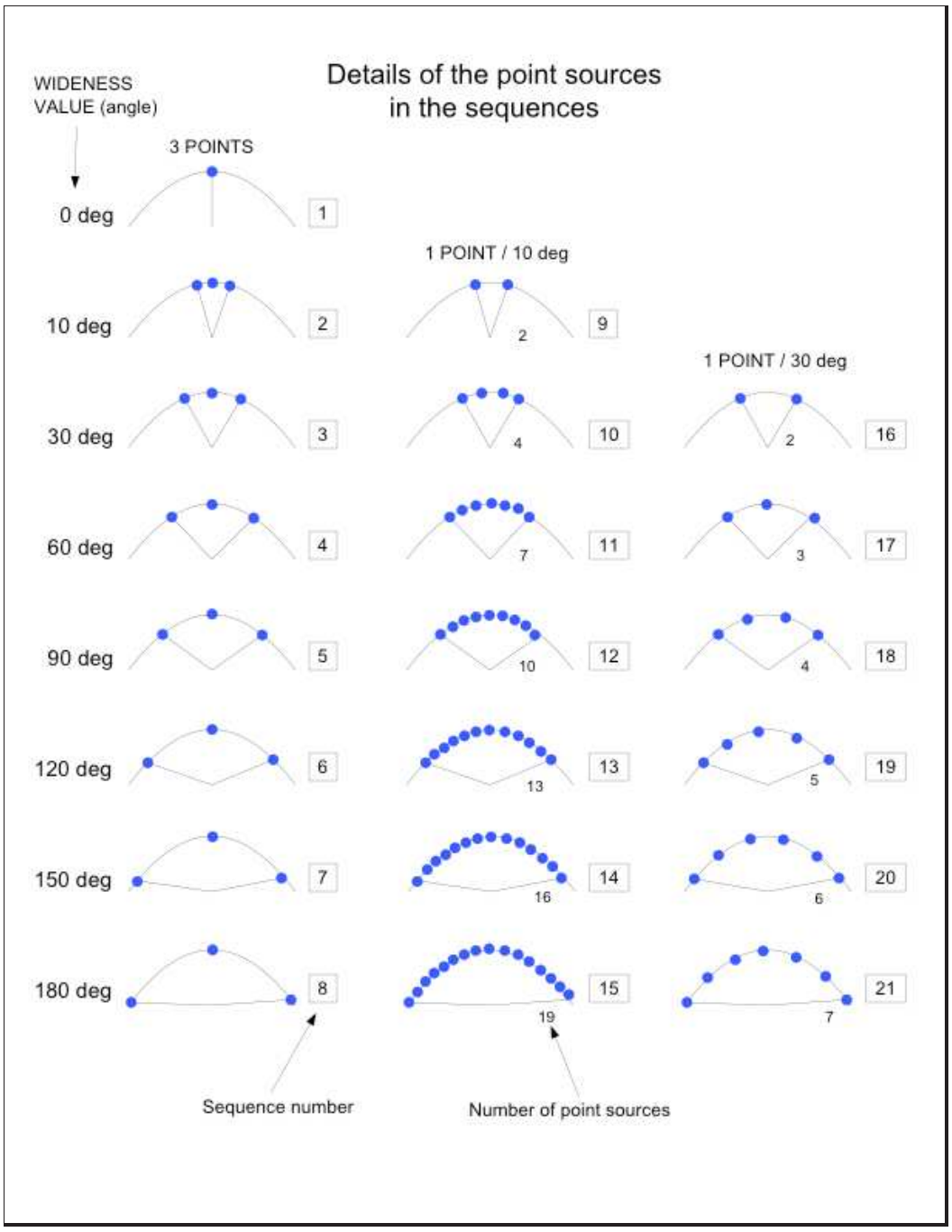


Figure 4.2: Construction of 21 horizontally extended sound source stimuli using three different densities of decorrelated point sources

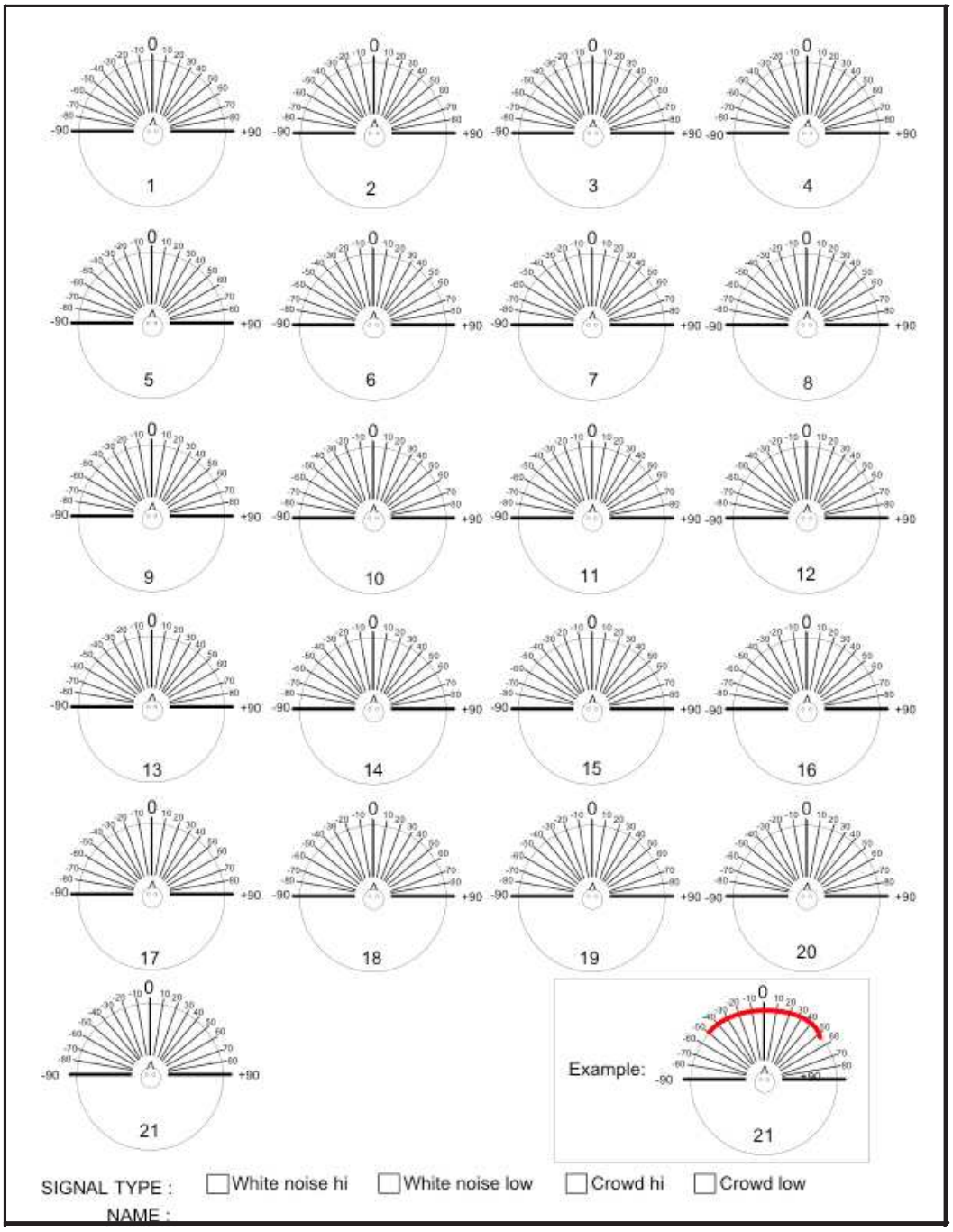


Figure 4.3: Answer sheet for drawing the perceived horizontal extents of the presented stimuli

4.3.5 Results

The arcs of circle drawn by subjects were used to compute the distribution of answers in the 0-180 degree range. This gave the average perceived sound source extent for each of the 84 stimuli. A value of 1 on the distribution graphs meant that 100 % of subjects drew the extent of the sound source at that location. The positions of the decorrelated sound sources used to create the stimuli are illustrated as small triangles for convenience. Results of the mean perceived extent of the 84 different stimuli are shown in Fig. 4.4 to 4.11. The type of signal used in the stimuli is indicated above the density graphs (WH : white noise high (0dB), WL: white noise low (-6dB), CH: crowd high (0dB), CL: crowd low (-6dB)).

Average results across the four signal types and for the 21 stimuli are shown in Fig. 4.12 and Fig. 4.18.

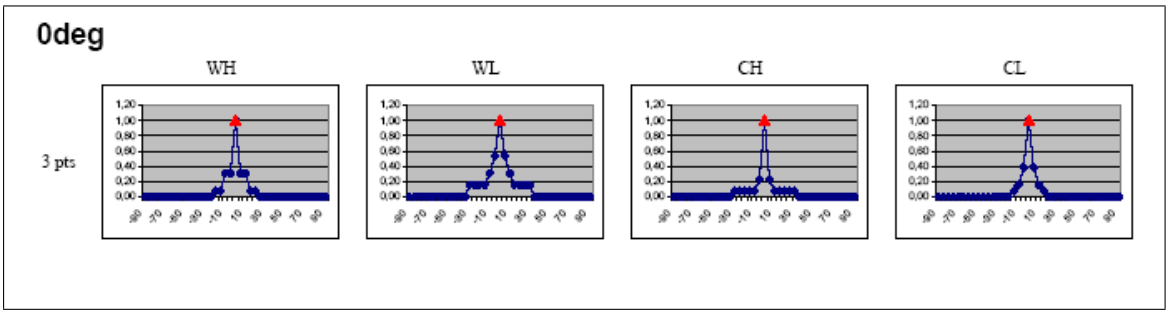


Figure 4.4: Mean perceived extent of 0 degree extended stimuli for four types of signals

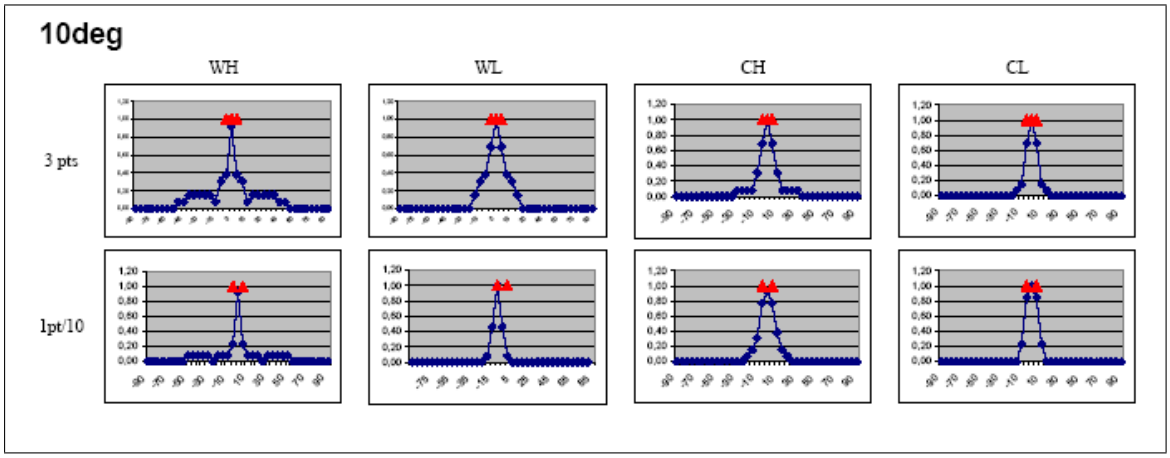


Figure 4.5: Mean perceived extent of 10 degree extended stimuli for four types of signals at two different point source densities

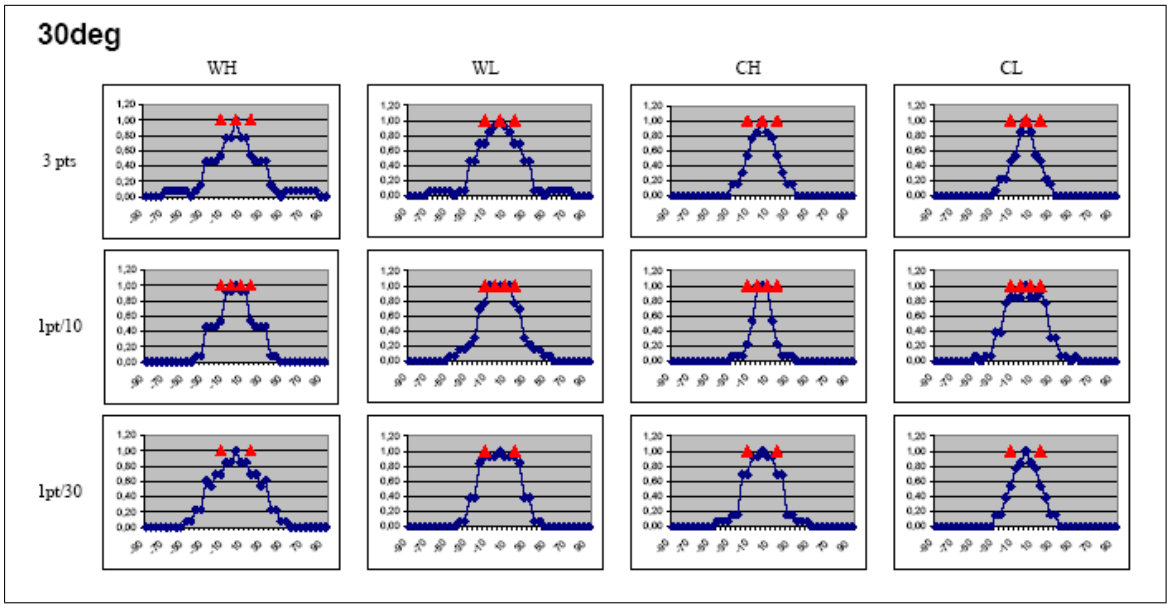


Figure 4.6: Mean perceived extent of 30 degree extended stimuli for four types of signals at three different point source densities

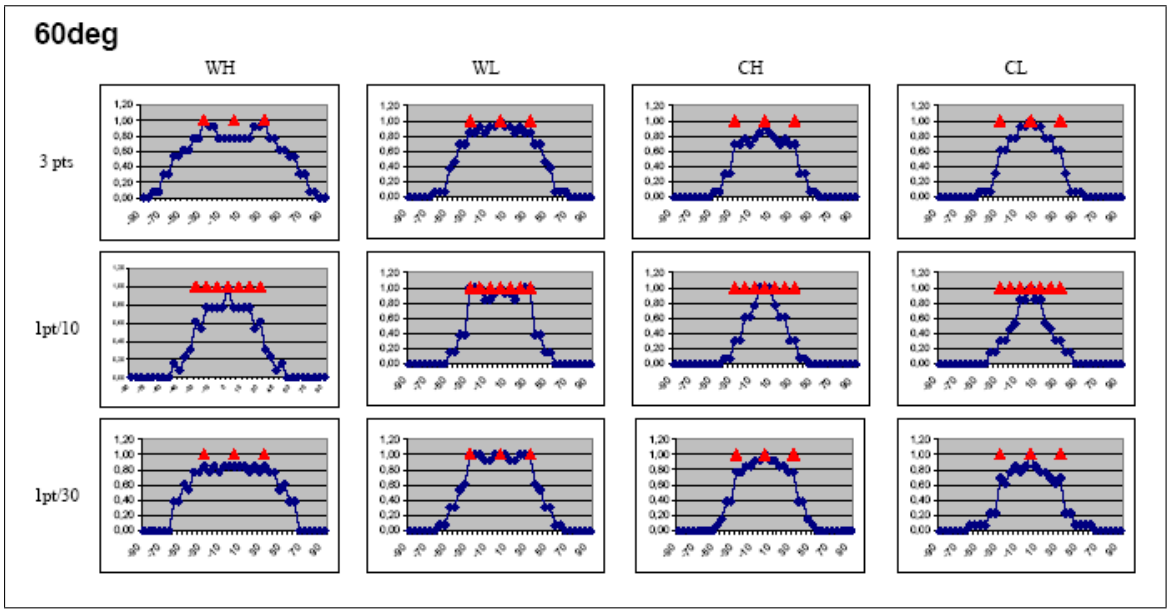


Figure 4.7: Mean perceived extent of 60 degree extended stimuli for four types of signals at three different point source densities

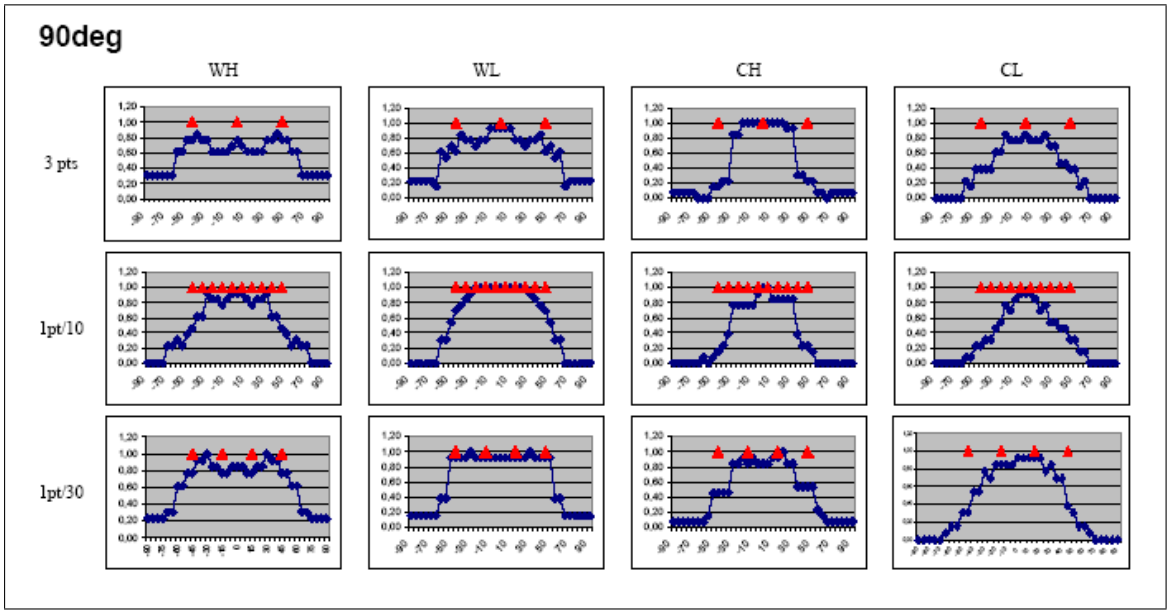


Figure 4.8: Mean perceived extent of 90 degree extended stimuli for four types of signals at three different point source densities

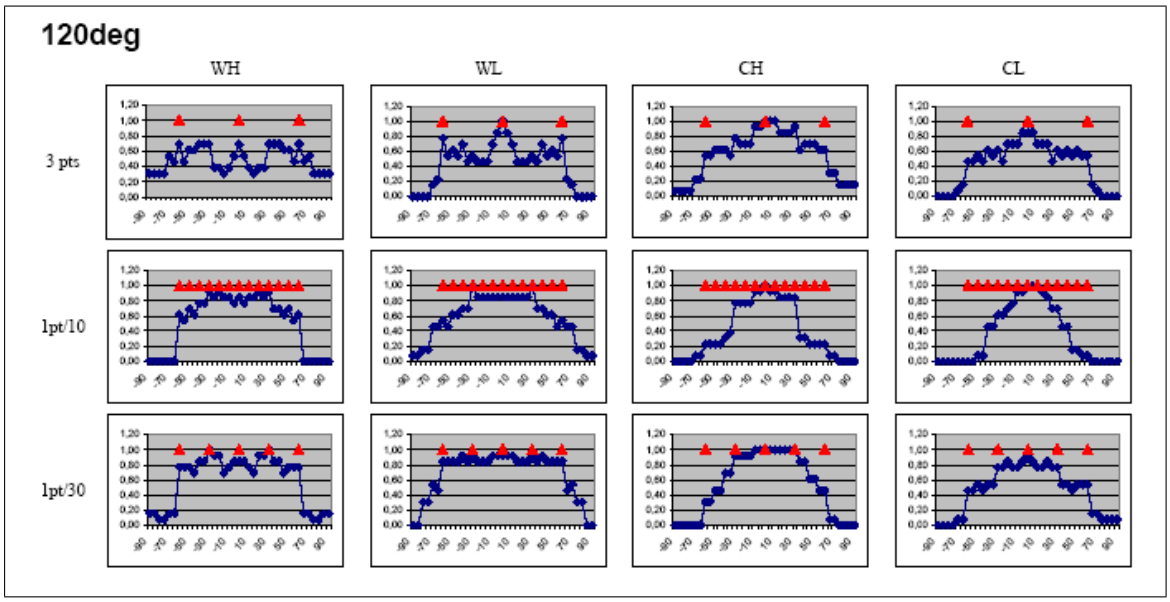


Figure 4.9: Mean perceived extent of 120 degree extended stimuli for four types of signals at three different point source densities

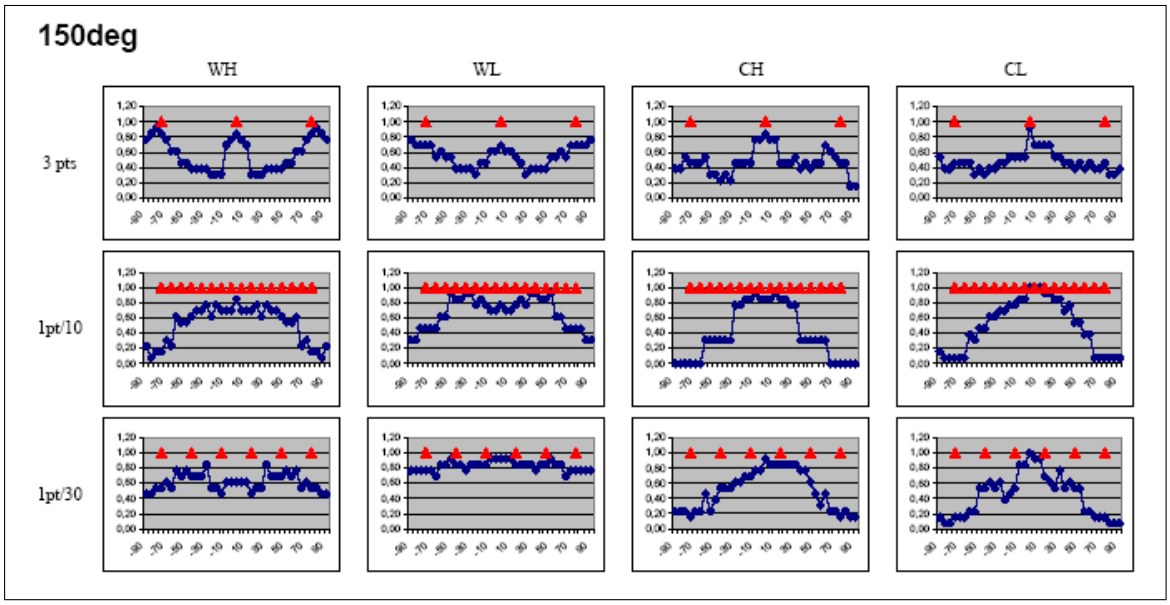


Figure 4.10: Mean perceived extent of 150 degree extended stimuli for four types of signals at three different point source densities

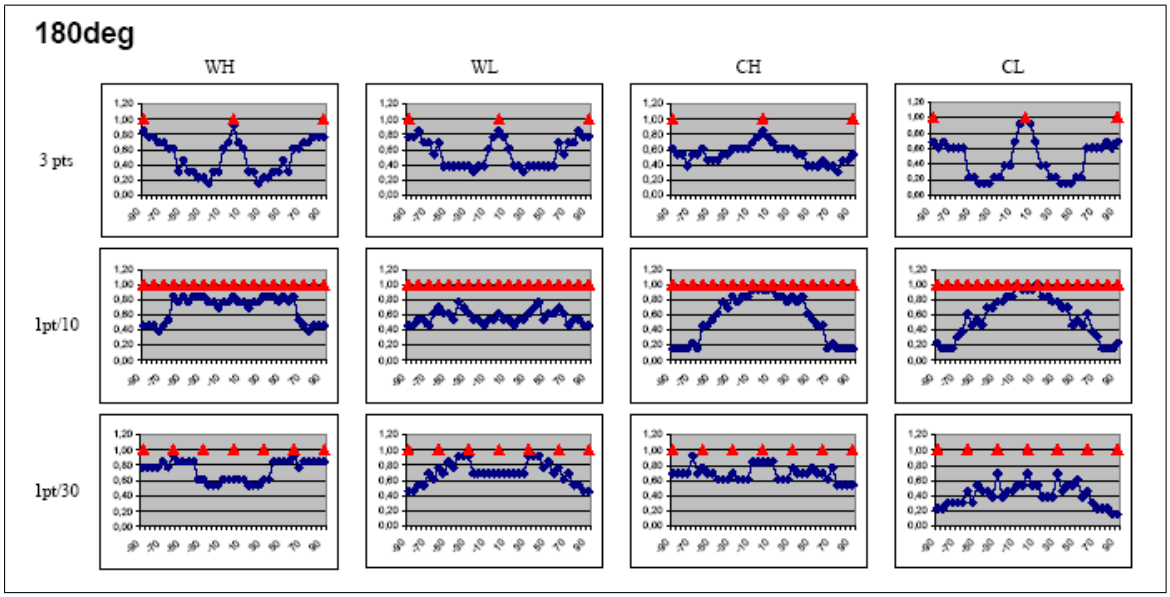


Figure 4.11: Mean perceived extent of 180 degree extended stimuli for four types of signals at three different point source densities

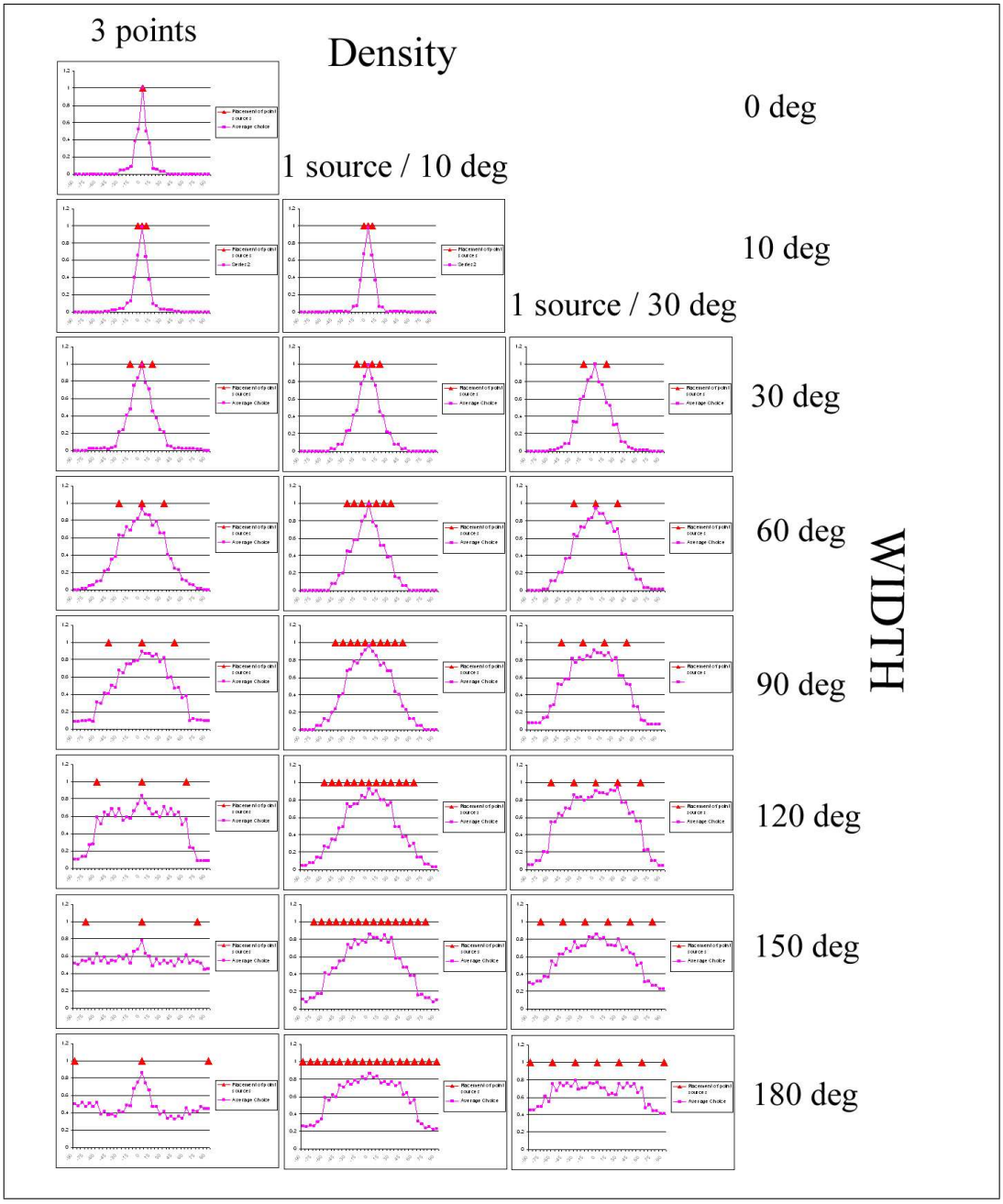


Figure 4.12: Mean perceived horizontal extent of the 21 stimuli across the four signal types

4.3.6 Analysis of Results

Mean errors and confidence intervals

Another way to analyse the subject answers is to compute the error between the source width transcribed by the subjects and the actual source width of the stimuli. The error is defined as (answered width − actual stimulus width). The mean error and 95% confidence interval were computed for each signal types (white and crowd noise at 0dB and -6dB) and for each sound source density (3 sources, 1 source per 10 degree, 1 source per 30 degree). Results are shown in Fig. 4.13, 4.14 and 4.15.

The average error and confidence interval across the four signal types were also calculated, these are shown for each source density in Fig. 4.16. The grand mean error and confidence intervals across the three source densities, the two signal types and the two signal levels are shown respectively in Fig. 4.17, Fig. 4.18 and Fig. 4.19.

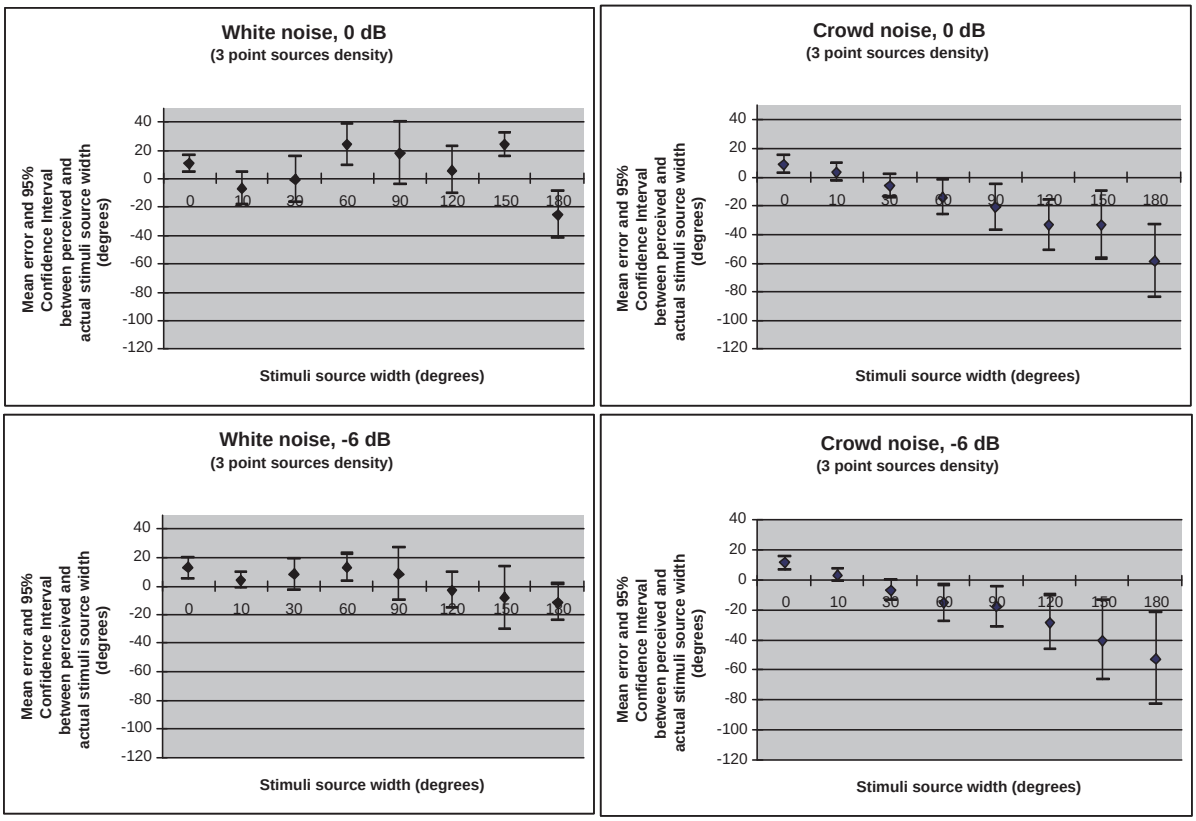


Figure 4.13: Mean Error and 95% confidence intervals between perceived and actual source width for 3 point source density

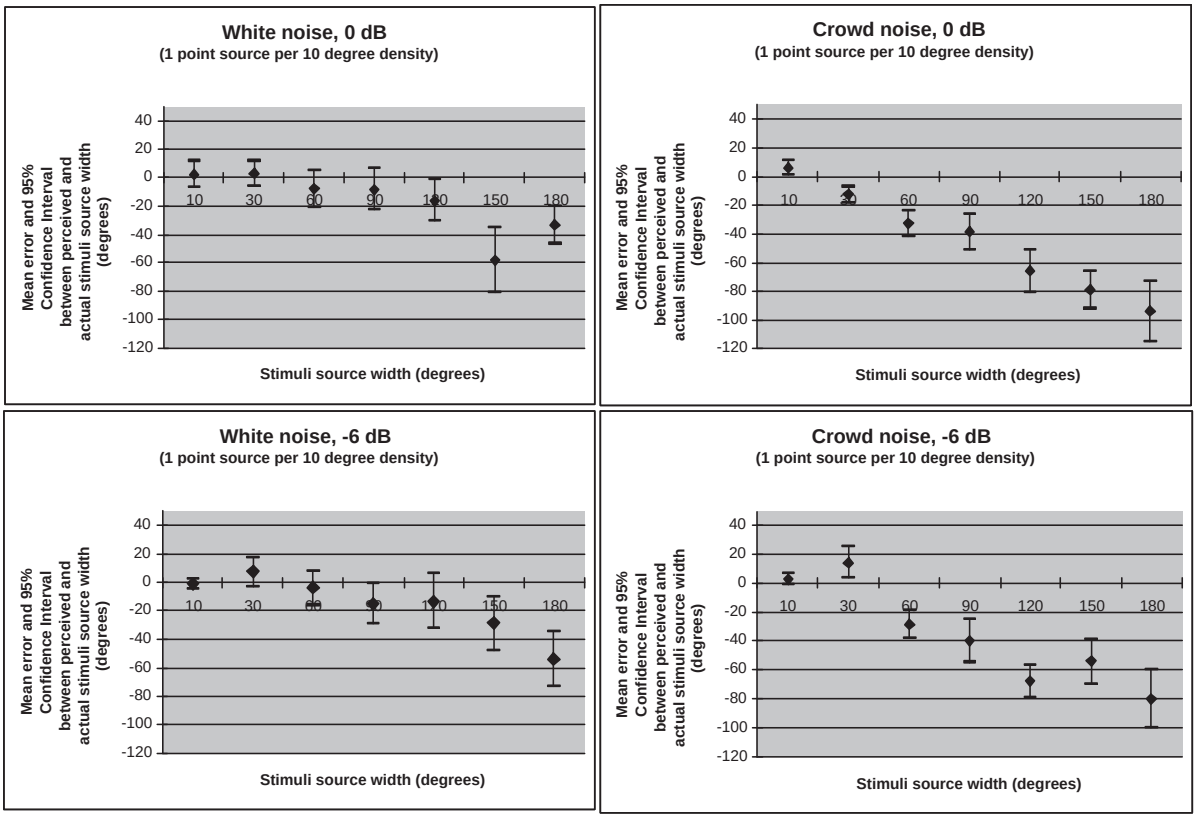


Figure 4.14: Mean Error and 95% confidence intervals between perceived and actual source width for one sound source per 10 degree density

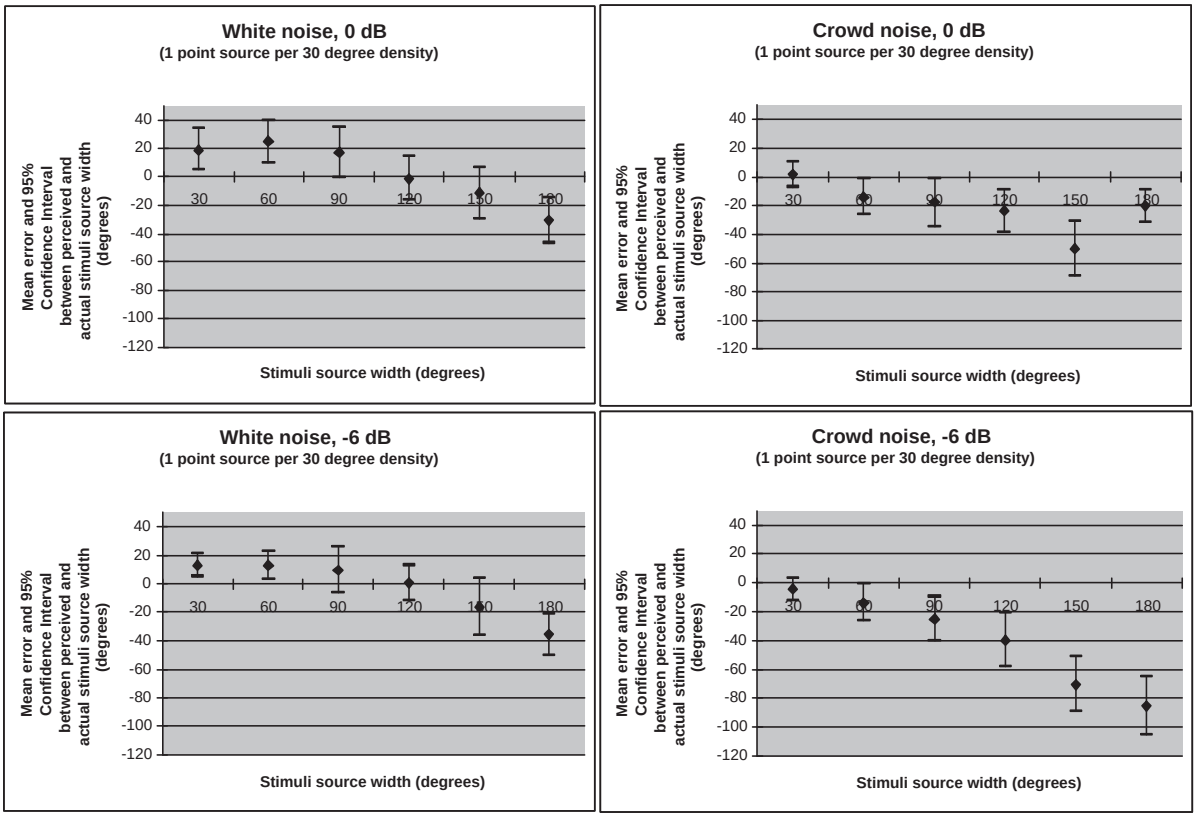


Figure 4.15: Mean Error and 95% confidence intervals between perceived and actual source width for one sound source per 30 degree density

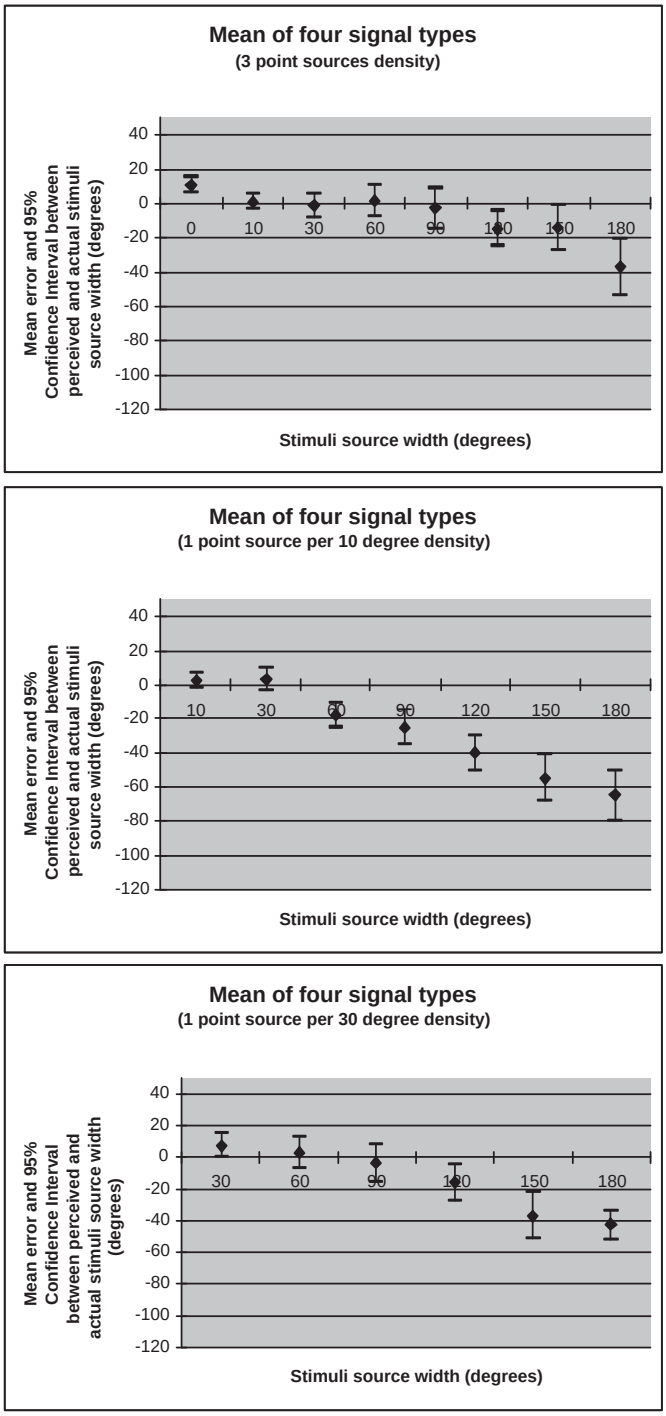


Figure 4.16: Mean Error and 95% confidence intervals between perceived and actual source width at three sound source densities

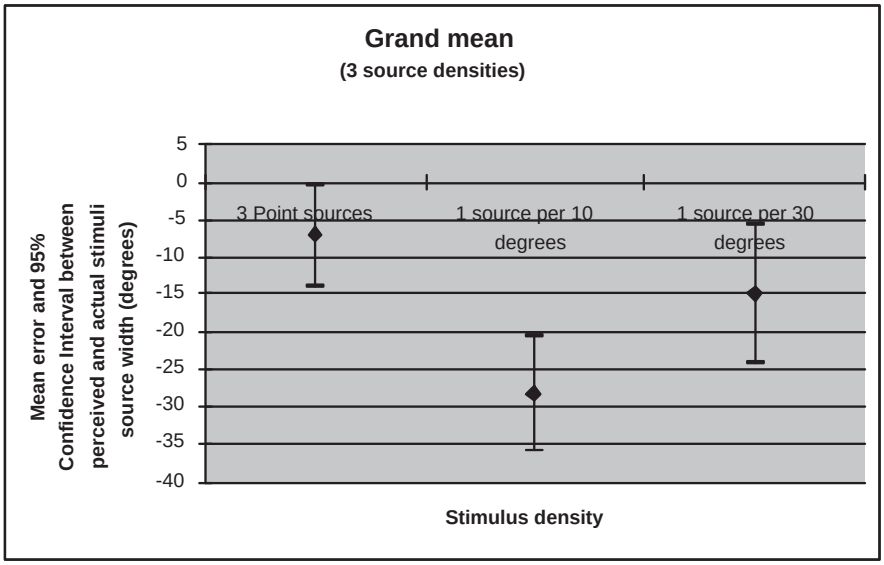


Figure 4.17: Grand mean error and 95% confidence intervals between perceived and actual source width at three sound source densities

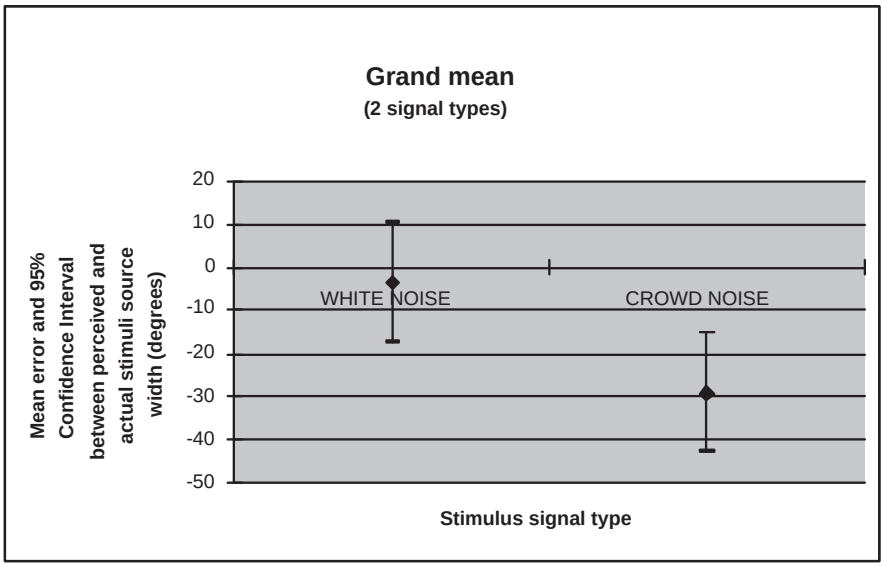


Figure 4.18: Grand mean error and 95% confidence intervals between perceived and actual source width for the two stimulus signal types

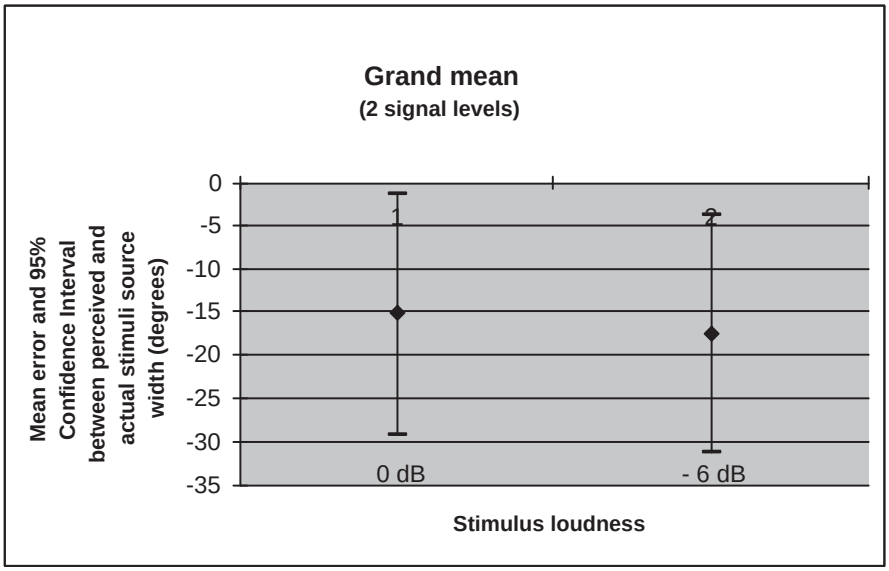


Figure 4.19: Grand mean error and 95% confidence intervals between perceived and actual source width for the two stimulus levels

ANOVA Analysis

The error between subject answers and actual stimuli width was also used to perform an ANalyis Of VAriance (ANOVA) [How02] using the SPSS statistical analysis software [Nor05]. For each sound source density a 3-factor ANOVA was carried out with the following factors: stimuli width, signal loudness (0 or -6 dB) and signal type (white noise or crowd noise). The ANOVA analysis gives F-ratios and 95 % confidence intervals for each factor and factor interaction, thus indicating which factor or factor interaction affected the error between the actual width of the presented stimuli and the widths reported by subjects. ANOVA results for the three sound source densities are shown in Fig. 4.20, 4.21 and 4.22. Confidence intervals smaller than 0.05, are marked with an asterisk; this indicates that the particular factor or factor interaction had a significant effect.

A four-factor ANOVA was also performed on subject answers for source widths ranging from 30 to 180 degrees; results for 0 and 10 degrees were not used so as to balance the analysis (since 0 and 10 degree stimuli were not present for all sound source densitities, see Fig. 4.2). Results of the four-factor ANOVA are shown in Fig. 4.23.

FACTOR	F	Sig.
WIDTH	8.264	0.000 *
LOUDNESS	0.301	0.589
SIGNAL TYPE	45.416	0.000 *
WIDTH * LOUDNESS	2.781	0.043 *
WIDTH * SIGNAL TYPE	9.851	0.000 *
LOUDNESS * SIGNAL TYPE	0.758	0.393
WIDTH * LOUDNESS * SIGNAL TYPE	2.227	0.088

Figure 4.20: 3-Factor ANOVA: F-ratios and confidence interval for 3 point source density

FACTOR	F	Sig.
WIDTH	26.958	0.000 *
LOUDNESS	4.638	0.042 *
SIGNAL TYPE	89.109	0.000 *
WIDTH * LOUDNESS	5.776	0.002 *
WIDTH * SIGNAL TYPE	15.358	0.000 *
LOUDNESS * SIGNAL TYPE	3.042	0.095
WIDTH * LOUDNESS * SIGNAL TYPE	5.812	0.002 *

Figure 4.21: 3-Factor ANOVA: F-ratios and confidence intervals for 1 source per 10 degree density

FACTOR	F	Sig.
WIDTH	28.202	0.000 *
LOUDNESS	26.172	0.000 *
SIGNAL TYPE	76.914	0.000 *
WIDTH * LOUDNESS	3.170	0.032 *
WIDTH * SIGNAL TYPE	6.171	0.002 *
LOUDNESS * SIGNAL TYPE	5.692	0.026 *
WIDTH * LOUDNESS * SIGNAL TYPE	3.620	0.019 *

Figure 4.22: 3-Factor ANOVA: F-ratios and confidence intervals for 1 source per 30 degree density

FACTOR	F	Sig.
DENSITY	79.344	0.000 *
WIDTH	24.146	0.000 *
LOUDNESS	1.809	0.192
SIGNAL TYPE	91.589	0.000 *
DENSITY * WIDTH	18.550	0.000 *
DENSITY * LOUDNESS	24.358	0.000 *
WIDTH * LOUDNESS	2.223	0.097
DENSITY * WIDTH * LOUDNESS	2.415	0.069
DENSITY * SIGNAL TYPE	0.260	0.774
WIDTH * SIGNAL TYPE	5.712	0.003 *
DENSITY * WIDTH * SIGNAL TYPE	1.594	0.213
LOUDNESS * SIGNAL TYPE	0.065	0.801
DENSITY * LOUDNESS* SIGNAL TYPE	6.750	0.005 *
WIDTH * LOUDNESS * SIGNAL TYPE	0.835	0.542
DENSITY * WIDTH * LOUDNESS * SIGNAL TYPE	8.197	0.000 *

Figure 4.23: 4-Factor ANOVA: F-ratios and confidence intervals

Perception of sound source unity

The subject results were then examined to study the case where some of the stimuli were not perceived as a continuous, single sound source (where subjects answers were not drawn as a single arc of a circle). This test was performed to study the causes for loss of binaural fusion[2] for some of the stimuli. Figure 4.24 shows the percentage of answers where the stimuli were perceived as single sound sources; details for the different stimuli widths and signal types are given in Fig. 4.25

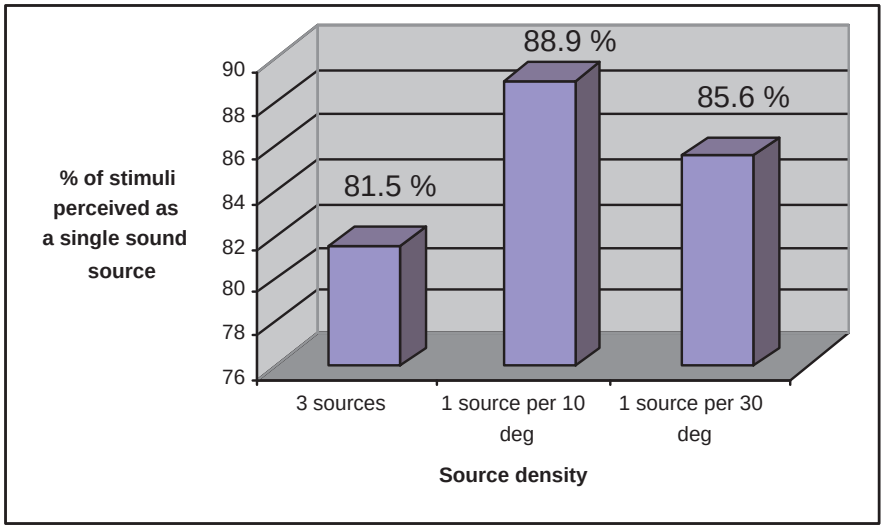


Figure 4.24: Mean of stimuli perceived as one sound source

[2]Binaural fusion was reviewed in section 2.8.5

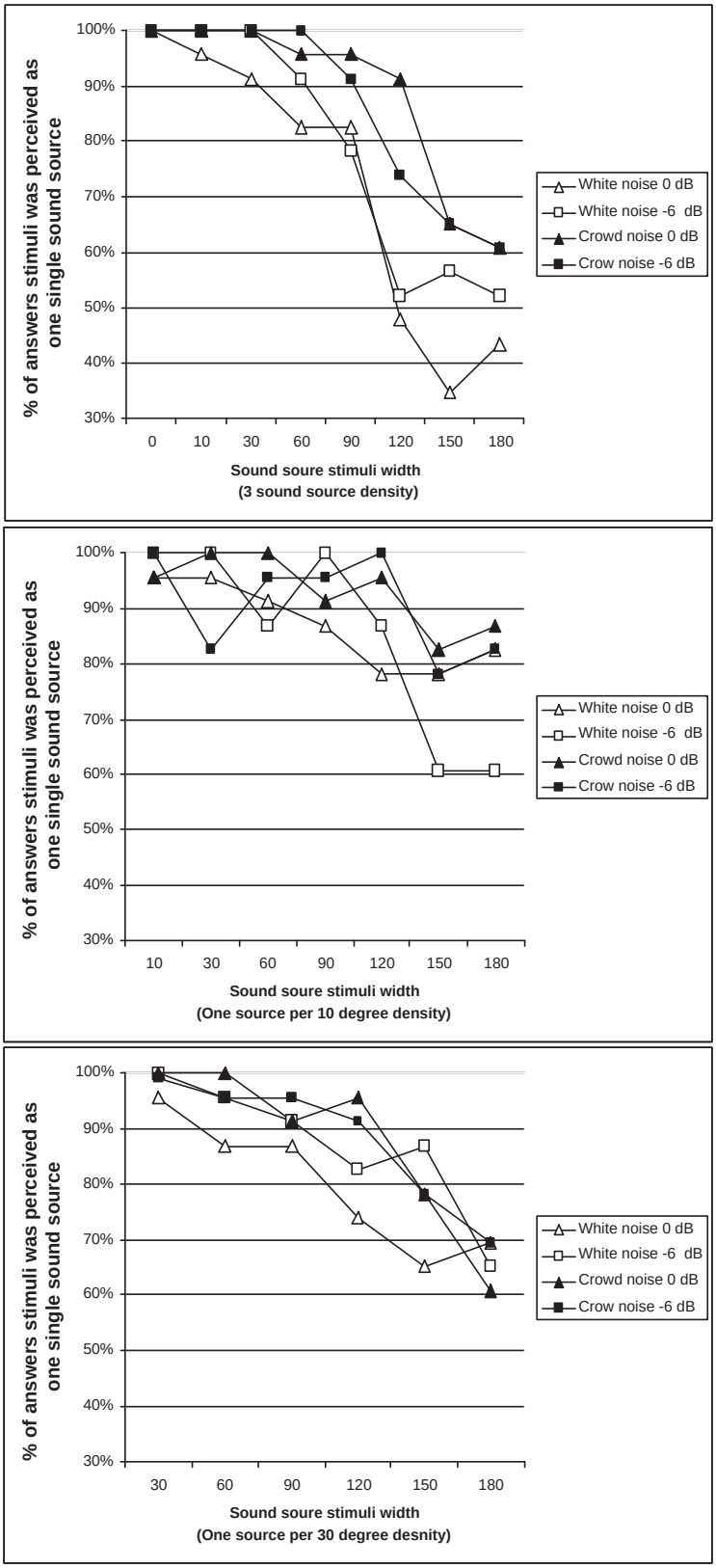


Figure 4.25: Percentage of answers where stimuli were perceived as single sound sources

4.3.7 Discussion

Effects of stimulus width and point source density

The error graph in Fig. 4.17 shows that sound source width perception was in general underestimated and, as Fig. 4.16 shows, more so as the width of the stimulus increased. A higher point sound source density also resulted in greater underestimation of the stimuli width (Fig. 4.17). The density factor had a significant effect on the error (and underestimation) with a relatively large F-ratio (Fig. 4.23: F-ratio= 79.344 and Sig=0.000). Underestimation of stimulus width due to high point source density (e.g. 1 source per 10 degrees) is also clearly visible in the mean subject answers shown in Fig. 4.12. Underestimation of stimulus width with high source density (and consequently with a high number of point sources) is due to a higher absolute value of the Inter-aural cross-correlation coefficient due to correlation between point sources introduced by the subject HRTFs[3]. This also explains that source width was more underestimated for wide source stimuli for the 1 source per 10 degree and 1 source per 30 degree densities and not for the 3 point source case (Fig. 4.16), as in the latter case the number of point sources remained constant. Thus the number of point sources used in the stimuli increased the absolute value of the IACC and in turn reduced the perception of source width; this finding is in line with the experiments of Kurozumi and Ohgushi [KO83] and Damaske [Dam68] described in section 2.8.4.

From the ANOVA tables shown in Fig. 4.20, 4.20, 4.21, 4.22 and 4.23, it can be seen that stimulus width was always a significant factor (Sig. < 0.05) on the error between transcribed and actual stimuli width. However, for the 3 point source density case, stimuli width had less impact on the error than with the other two densities (F-ratio of 8.264 instead of 26.958 and 28.202 respectively); this observation is also visible in Fig. 4.16. This finding also relates to the previous observation that higher source density increased the number of point sources for wide sources and tended to

[3]The link between Inter-source correlation coefficient (ISCC) and the Inter-aural cross-correlation coefficient (IACC) was studied in section 2.8.3

narrow the perceived extent of the stimuli. However, it does not explain why the 3 point source case (with constant number of sources) also presents underestimation for increased source extent (to a lesser extent as shown by a lower F-ratio). In this case, underestimation is due to other mechanisms such as perception of the stimuli as multiple sound sources due to binaural fusion effects; this is studied shortly below.

Increased stimulus width and density thus interacted towards narrowing the perceived source extent; this is reflected by F-ratio= 18.550 and Sig=0.000 for the (Density*Width) interaction factor in Fig. 4.23.

On the contrary, further interactions of (Density*Width) with signal loudness or signal type were not significant (Sig.= 0.069 and 0.213 respectively in Fig. 4.23).

Effects of signal type

The signal type used in the stimuli (white noise or the sound of a large crowd) was the most significant factor influencing the error between transcribed and actual stimuli width as reported by the highest F-ratios in the ANOVA tables shown in Fig. 4.20 to 4.23. Indeed, the crowd noise signal increased underestimation of the stimuli width; this can be seen in Fig. 4.18. Reasons for this effect are not entirely clear but could be linked to the fact the crowd noise had less spectral energy than white noise (which has a constant energy at all frequencies) and this in turn can alter the perception of tonal volume which was reviewed in section 2.7.

From the ANOVA tables of Fig. 4.20 to 4.23 it is also seen that signal type and stimulus width interacted significantly (Sig. < 0.05), meaning that for the crowd noise signal, underestimation was worsened as the width of the stimuli increased. From the ANOVA tables it can be seen that signal type did not interact with signal loudness in a significant way except for the 1 source per 30 degree density case (Fig. 4.22), however in this case the F-ratio is low, indicating a weak effect.

Signal type did not interact with source density in a significant manner (Sig. $>$ 0.05) and weakly interacted (F-ratios low) with (density * loudness) and (density * width * loudness) as shown in the ANOVA table of Fig. 4.23.

Effects of signal loudness

Stimulus loudness (0 or -6 dB) did not significantly affect the error between transcribed and actual stimuli width as reported by significance values greater than 0.05 in all ANOVA tables (Fig. 4.20 to 4.23). This observation is also visible in Fig. 4.19 where the stimulus level slightly reduced the grand mean error between transcribed and actual stimulus width.

On the other hand, stimulus loudness and density interacted quite significantly as reported by a F-ratio of 24.358 in ANOVA table of Fig. 4.23.

Stimulus loudness also interacted significantly with stimulus width (but weakly) as reported by significance values greater than 0.05 and relatively low F-ratios for (Width * Loudness) in ANOVA tables of Fig. 4.20, 4.20, 4.21.

Perception of single or multiple sound sources

Subject answers were also analysed to study the case where the presented stimuli were not perceived as single, continuous sound sources. Graphs in Fig. 4.25 shows the percentage of stimuli that were not perceived as single sound sources, in function of stimulus width. From this graph, it is apparent that the width of the stimulus decreased the percentage of answers where the stimulus was perceived as a single sound source. This observation applies to all signal types, levels and source densities.

The lowest source density (3 point sources) produced the lowest percentage of answers perceived as continuous sound sources. This is corroborated by some of the distribution graphs of subject answers for large source width (> 100 degrees) and low density (3 point sources) which show holes between the decorrelated sources; this can be seen for instance in Fig. 4.10 and Fig. 4.11. This indicates that the phenomenon of binaural fusion detailed in section 2.8.5 was lost due to decorrelated sound sources being too far apart.

White noise stimuli also resulted in a slightly lower percentage of answers being perceived as continuous sound sources at the three source densities (Fig. 4.25). This can be explained by the fact that the crowd noise tended to be more easily perceived

as a single auditory event than white noise due to cognitive grouping (section 2.8.5).

Except for the lowest density case (3 point sources) with white noise, subjects perceived the stimuli as continuous sound sources more than 60% of the time for all signal types and levels (Fig. 4.25).

The averages of answers where subjects perceived the stimuli as continuous sound sources for the three source densities are shown in Fig. 4.24, it can be seen that higher source density increases the likelihood of the stimulus being perceived as a single sound source.

General conclusions

The present experiment showed that it is possible to artificially render sound sources such that they are perceived as being horizontally extended. The stimuli that produced the best match between intended and perceived source extent were those which employed a 3 point source density and if the sound source width did not exceed 150 degrees (Fig. 4.16 and 4.17).

Source density was a statistically significant factor affecting the error between intended and perceived source width. It appeared that too high the density of point source distribution resulted in underestimation by subjects of the stimuli width, especially as the number of point sources increases with the width of the stimuli. In contrast, too low the density resulted in loss of binaural fusion, and subjects individually perceived the point sources constituting the broad sound source. Therefore, when rendering source extent in virtual auditory displays, the density parameter must be selected as a compromise. Another possible solution is to use a variable source density (such as in the three point source density stimuli) and to increase the number of point sources as the width of the sound source increases so as to avoid loss of binaural fusion.

From an ANOVA analysis it was found that signal loudness alone did not significantly affect the perception of source extent in the 0 to -6 dB range. Signal loudness however interacted significantly with stimuli width. Signal type (white or crowd

noise) was a highly significant factor. The variations of the error between intended and perceived source width due to the nature of the sound source audio signal indicate that when using the decorrelated point source technique to render source width, a particular sound source width cannot be guaranteed for all signal types. A more advanced source extent rendering scheme could analyse[4] the sound source signal and modify the position of the sound sources so as to compensate adverse effects of the source signal.

In conclusion, this experiment showed that under certain conditions the rendering of horizontal sound source extent is possible with the decorrelated point source technique and even, with high precision. This good precision can be used in the process of data sonification [Kra94] where the extent of sound sources could be used to convey information. The apparatus used to perform the experiment, however, did not reflect a real speaker array used in 3D auditory systems, since it only allowed horizontal broad sound sources to be produced and at the front of listeners. The topic of the next experiment described in section 4.4 is to study the perception of artificial sound source extent on a more realistic speaker array which permits full periphonic[5] sound and thus, permits creating one and two-dimensional extended sound sources to be placed anywhere around subjects.

4.4 Experiment 2: perception of horizontal, vertical and 2D sound source extent

4.4.1 Aims

In this experiment, the perception of rendered source extent in the context of a real 3D audio system is explored. Unlike the apparatus of the previous experiment (section 4.3), this experiment utilises a periphonic 3D audio rendering system which

[4] by measuring subjective loudness and spectral energy for instance

[5] That is, coming from all directions

allows placing sound sources anywhere around the listener. Using this apparatus, the perception of rendered sound source extent is studied for stimuli having horizontal and vertical extents and two-dimensional rectangular extents. The aims of this experiment are to study the precision by which subjects are able to perceive these different extents and to study if the different stimuli types (i.e. horizontal, vertical, rectangular) are indeed perceived differently. Another aim is to study the impact of the position (i.e. azimuth and elevation) of the stimuli on the ability by subjects to perceive the extent of sound sources.

4.4.2 Apparatus

The experiment was carried out in the Configurable Hemispheric Environment for Spatialised Sound (CHESS) which is a novel system described in chapter 5. This apparatus consists of a 16-speaker array placed in a geodesic dome configuration. The coordinates of the speakers and geometry of the array are shown in Fig. 4.26 and is further detailed in section 5.4.1. Subjects were placed at the centre of the speaker array and were facing the point (azimuth=0, elevation=0) as shown in Fig. 4.27.

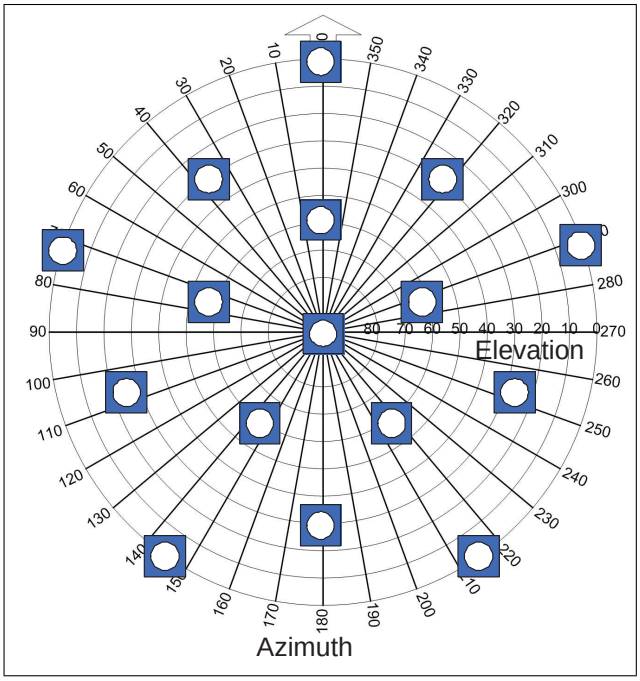


Figure 4.26: Geometry of the 16-speaker array apparatus

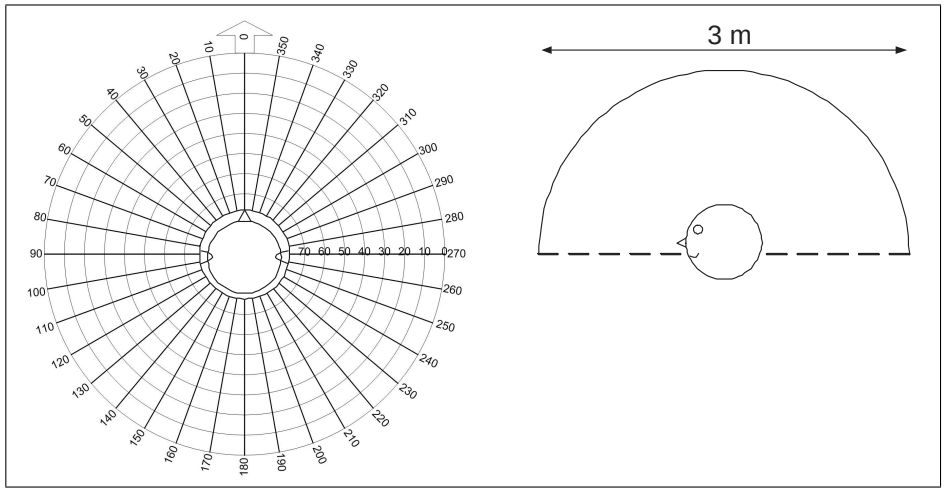


Figure 4.27: Position of the subjects in relation to the apparatus

4.4.3 Stimuli

To create stimuli with various spatial extents, the decorrelated point source technique described in section 2.11 was used. The stimuli were constructed using six point sources emitting independent (i.e. uncorrelated) white noise signals. Using 4th order Ambisonics[6] spatialisation to place the decorrelated sound sources on the speaker array, sixteen extended sound source stimuli having various extent geometries and positions were obtained. These are represented in Fig. 4.29 by a thick line and consist of:

- Four horizontally extended sound sources with an extent of 60 degrees (sequences 1 to 4)
- Four horizontally extended sound sources with an extent of 180 degrees (sequences 11 to 14)
- Four vertically extended sources with extents of 40 and 90 degrees (sequences 5 to 8)

[6]This spatialisation technique is detailed in section 5.4.1

- One rectangular (2D) extended sound sources with an horizontal extent of 60 degrees and a vertical extent of 30 degrees (sequence 10)
- One rectangular (2D) extended sound sources with an horizontal extent of 180 degrees and a vertical extent of 40 degrees (sequence 9)
- Two point sound sources (sequence 15 and 16)

4.4.4 Procedure

The sixteen broad sound source stimuli depicted in Fig. 4.29 were played to subjects in random order. For each presented stimuli, subjects were asked to draw the perceived extent of the stimuli. An answer sheet that represented a top-down view of the speaker dome array was used (Fig. 4.28); the centre of the answer sheet thus represented the zenith of the dome. Subjects were allowed to draw lines or any shape to transcribe the perceived extent of the different stimuli. Although prone to some transcription errors, this elicitation method was judged to be appropriate for transcribing the perceived 1D and 2D sound source extents by subjects.

Subjects were placed at the centre of the dome and were facing the zero degree orientation represented in Fig. 4.28 by an arrow. Head rotations were allowed and subjects were not visually masked. Fifteen subjects with normal hearing and no particular knowledge in the audio field participated in the experiment. Subjects were postgraduate research students from the University of Wollongong and there was an approximately equal number of male and female participants.

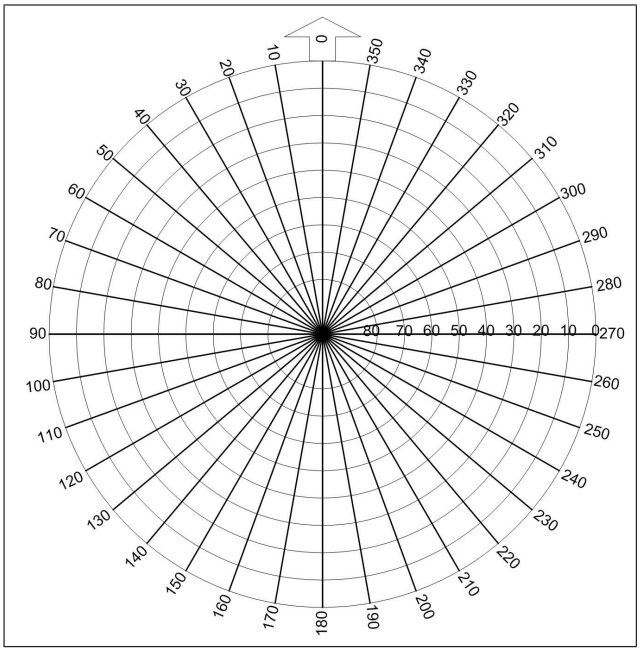


Figure 4.28: Answer sheet for the 2D sound source extent experiment

4.4.5 Results

Areas where subjects had drawn were counted and from this data, distribution density graphs were obtained; these are shown for each of the sixteen stimuli in Fig. 4.29. The density graphs thus represent the mean perceived two-dimensional source extent for each of the sixteen stimulus. The answers that were on-target were also counted, and the average of the answers calculated; these are also shown in Fig. 4.29. A mean on-target percentage of 100% means that a 100% distribution density (represented in black) completely covered the extent of the stimuli (but could extend beyond the area of the stimuli). Thus a high on-target percentage indicates that subjects perceived the extent of the stimuli where the stimuli was defined.

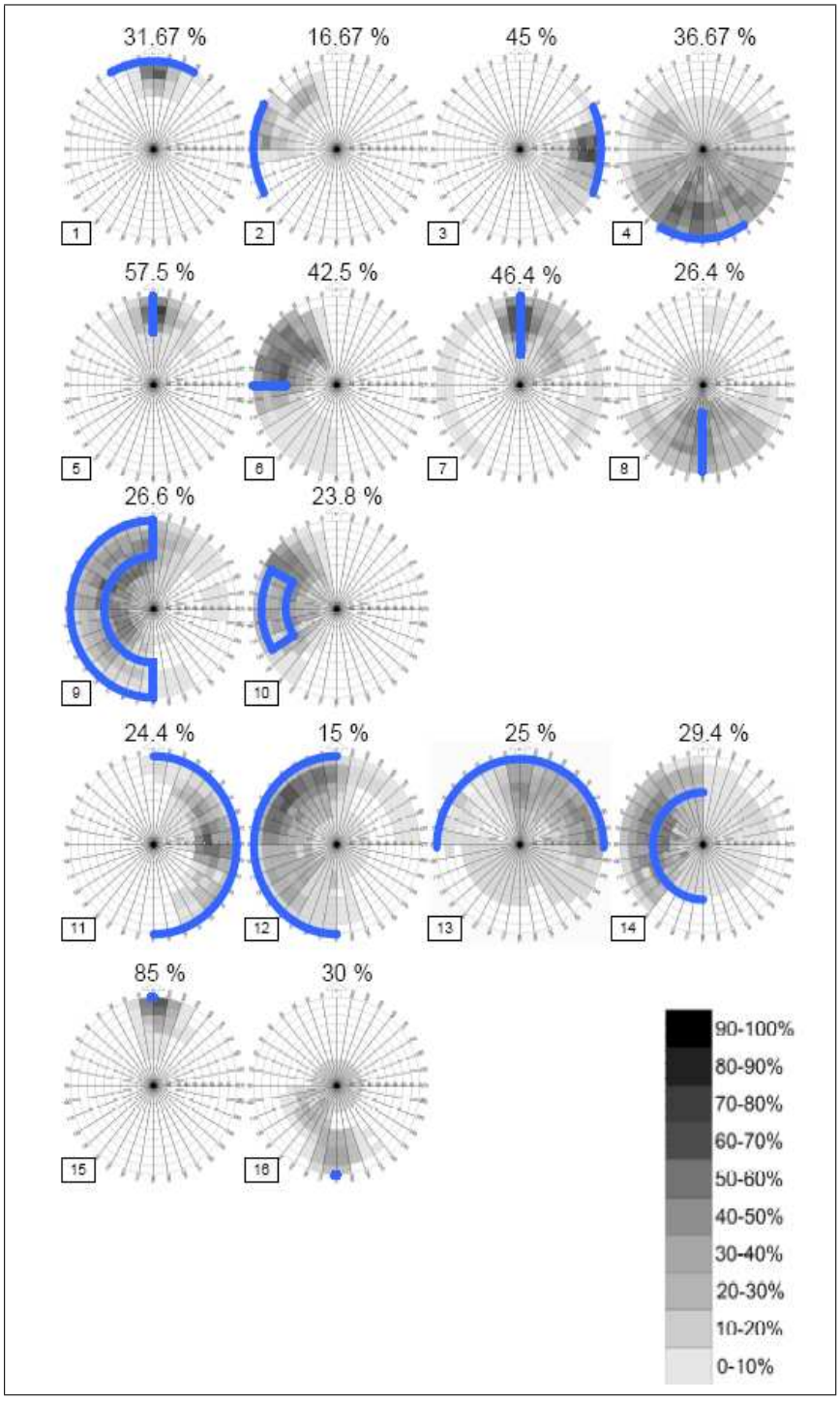


Figure 4.29: Distribution of perceived source extents and mean percentages of on-target answers for 1D and 2D sound sources presented on a three-dimensional auditory display

4.4.6 Discussion

The distribution density graphs of Fig. 4.29 show that, in general, the mean perceived source extent matched coarsely the intended sound source extent (thick line). For sources with an horizontal extent of 60 degrees (sequences 1 to 4), the perceived source extent was narrower than intended. This can be explained by the point source density being too high (one source per 15 degrees)[7] as discovered in the previous experiment (section 4.3). Indeed, this effect was not apparent with horizontal extents of 180 degrees (sequences 11 to 14), corresponding to a lower density of one source per 36 degrees[8]. However, for sequences 11 to 13, subjects perceived some elevation in the sound sources which was not intended; this could be linked to spatialisation errors of the Ambisonics technique. Sequence 14, which is an 180 degree horizontal sound source placed at 40 degree elevation was perceived as having more elevation than sequence 12, but not with a great precision however.

The perception of one-dimensional vertical sound sources (sequences 5 to 8) matched the intended source extent relatively well, provided that localisation in the vertical plane is worse than in the horizontal plane [Bla97]. By comparing results between sequence 5 and 7, it can be seen that differences in vertical source extent were perceived. Another comparison which can be made is between sequence 5 and 6 and between 7 and 8. For these, it is apparent that the perception of source extent on the side (sequence 6) and behind subjects (sequence 8) was less accurate than at the front (sequence 5 and 7). This finding hints that the precision by which listeners perceived sound source extent was positively influenced by localisation accuracy[9]. Indeed, this seems normal since subjects relied on the position of the decorrelated point sources to perceive the apparent extents of the stimuli.

For sources with a rectangular 2D extent (sequence 9 and 10), the intended and perceived extent matched well. Subjects also correctly indicated that these sources exhibited more vertical extent compared to line sound sources (sequence 2 and 3).

[7] Since 6 point sources were used, there are five equal intervals between them so that $\frac{60}{5} = 15$ deg

[8] $\frac{180}{5} = 36$ deg

[9] It is well known that localisation accuracy is best at the front [Bla97]

Thus, subjects could differentiate between sound sources with one-dimensional and two-dimensional extents.

From sequences 15 and 16, it can be seen that even a single speaker was not perceived as an absolute point source and had some spatial extent. This can be explained by the non-zero physical size of the speaker and by the tonal volume phenomenon which was reviewed in section 2.7.

Mean on-target answers were better for frontal (sequences 1,5,7 and 15) than rear presented stimuli (sequences 8 and 15); this is also related to localisation accuracy being more precise at the front that at the back. An exception is sequence 4 which has a higher mean on-target percentage than sequence 1. This is however due to the fact that the perception of sequence 4 was diffuse and blurry as shown in the distribution of answers. Mean on-target percentages were higher when the stimuli were presented to the right (sequence 3 and 11) than to the left (2 and 12). Reasons for this can be related to imperfections of the experiment room which was not anechoic and sound reflections may have biased sound source localisation.

In general, it can be concluded that the mean perceived spatial source extent matched coarsely the intended extent. However, elicitation errors are likely to have blurred the results of the mean perceived extent shown in Fig. 4.29. A more precise elicitation method such as using a laser pointer to draw the perceived source extent would likely provide tighter results.

Despite these elicitation errors, it can be seen that subjects were able to successfully perceive the different types of sound sources (horizontal and vertical line sources, rectangular sources and point sources), their sizes and their positions. The author suggested in [PB04a] that this ability could be used in data sonification applications. For example, in a 3D audio air-traffic control system, the extent and geometry of sound sources could be used to represent the position, size and distance of surrounding planes.

4.5 Experiment 3: perception of sound source shape using real decorrelated sound sources

The perception of horizontal sound source extent has been studied in section 4.3 and the perception of horizontal, vertical and rectangular (two-dimensional) sound source extents in section 4.4. This experiment now studies the ability of listeners to identify two-dimensional sound source extents exhibiting particular *shapes*. To do so, the decorrelated point source technique described in section 2.12 was used in a novel way to form apparent source shapes; that is, by placing the decorrelated sound sources in particular patterns, sound source shapes were obtained.

In this experiment, real decorrelated point sources (i.e. speakers) were used to create the sound source shape stimuli. Another experiment described in section 4.6 studies the case where virtual (i.e. spatialised) decorrelated point sources are used. The use of virtual decorrelated point sources reflects more the practical case of a 3D audio rendering system. In this experiment however, real decorrelated point sources are used so as to maximise the localisation and stability of point sources. This allows the experiment to focus on shape perception alone and avoids errors that are introduced by spatialisation inaccuracies etc.

4.5.1 Aims

This experiment studies the ability of humans to perceive different sound source shapes that have been artificially created. Unlike the experiment described in section 4.4 where subjects had to draw the perceived extent of sound sources, this experiment consisted of an *identification* task between six sound source shapes. The experiment was repeated for four types of signals and for frontal and rear presentations of the sound source shape stimuli so as to study effects of signal type and localisation accuracy on source shape perception.

This study was first conducted for the MPEG Audio group to study the need of

implementing sound source shape capabilities in the MPEG-4 standard (see section 4.10 for details). The experiment was designed by the author and was carried out at three locations: University of Wollongong, Thomson Multimedia (Germany) and ETRI (Korea). The participation of Jens Spille[10] and Jeongil Seo[11] for repeating this experiment at their respective laboratories is acknowledged.

4.5.2 Apparatus

The apparatus consisted of a 7-speaker array placed equidistantly at 1.6m of subjects (Fig. 4.30); the arrangement of the speakers is depicted in Fig. 4.31. The speakers were placed so that, from the subject's point of view, the array had a maximum horizontal extent of 80 degrees and a vertical extent of 43 degrees. Subjects were asked to face the central speaker of the array. The polar coordinates of the speakers are shown in Fig. 4.32. Pictures of the experiment apparatus at the three locations where the experiment was carried out are shown in Fig. 4.33.

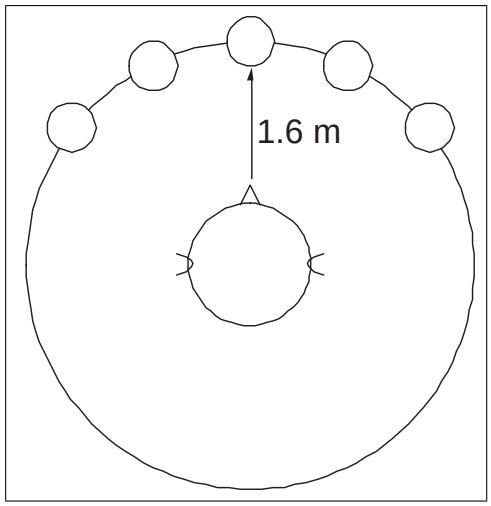


Figure 4.30: Position of the speaker array in relation to the subjects

[10]Research Engineer at Thomson Multimedia, Germany
[11]Research Engineer at ETRI, Korea

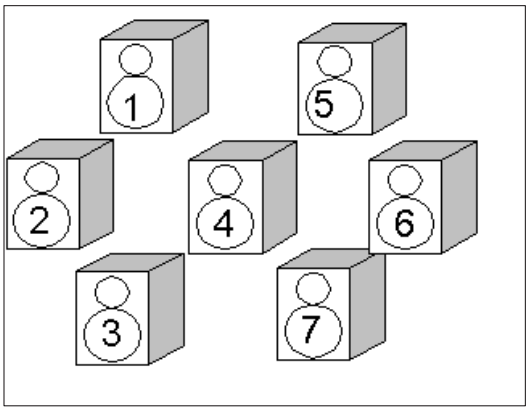


Figure 4.31: Diagram of the speaker array

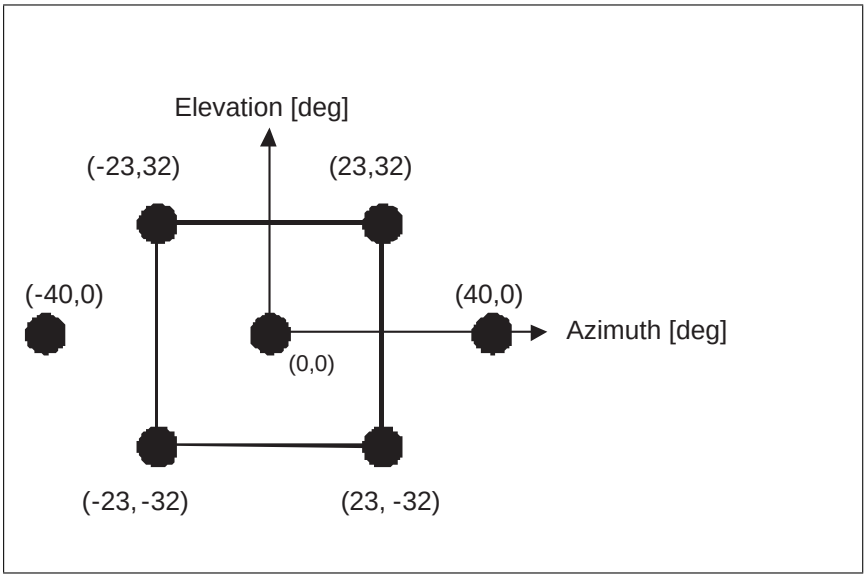


Figure 4.32: Coordinates of the decorrelated point sources/speakers

Figure 4.33: Apparatus of the sound source shape perception experiment with real decorrelated sound sources. From left to right: at Thomson (Germany), University of Wollongong and ETRI (Korea)

4.5.3 Stimuli

By switching on or off speakers of the array, six different sound source shapes were created as shown in Fig. 4.34. The signals fed to 1, 3, 5 or 7 loudspeakers were uncorrelated. Four types of signals were employed: white noise, 1kHz low-pass filtered noise, 3kHz high-pass filtered noise and a blues guitar riff. To obtain uncorrelated noise signals, independent noise sequences were used. To obtain seven uncorrelated signals for the blues guitar recording, a 256-tap FIR decorrelation filterbank[12] was used. Listened individually, there was no audible perceptual difference between the decorrelated signals. Loudness of the different shapes was normalised so that shape 'A' (one source) had the same loudness as shape 'F' (seven sources) so as to prevent subjects from using loudness as a cue to identify the source shapes. The shapes had the same centre of symmetry so as to prevent subjects from using localisation cues to identify the source shapes.

Stimulus sequences were created in which each of the six sound source shape was used three times and for each of the four signal types. In all, 72 ($6 \times 4 \times 3$) sequences were created and their order randomised. The duration of each sequence was 5 seconds.

[12]The implementation of FIR decorrelation filterbanks was described in section 2.13.2

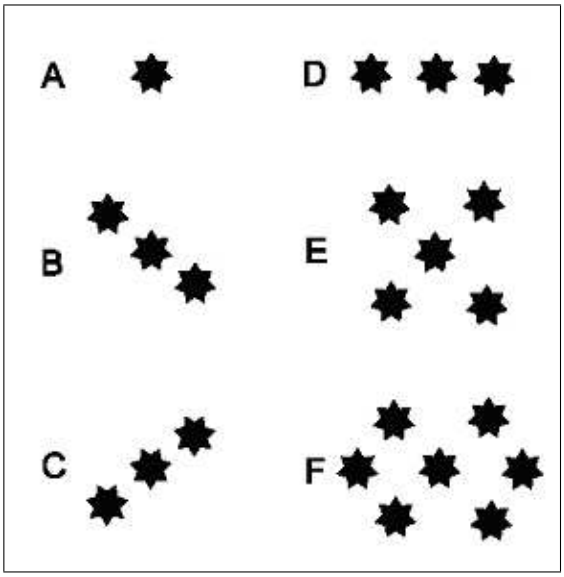


Figure 4.34: Geometry of the six sound source shapes used in the experiment

4.5.4 Procedure

Before starting the experiment, subjects were given training so that they were introduced to each sound source shape for each of the four signal types. After the training session, subjects were asked to identify among six possible shapes, the shape of each of the 72 presented sequences. Subjects could leave a blank answer if they could not identify a shape for a particular sequence. No feedback was given to subjects during the experiment. Subjects could rotate their heads freely, had no visual masking and could demand sequences to be repeated. The whole experiment was then repeated with the speaker array placed in the *back* of subjects. Between 26 and 10 adult subjects with normal hearing took part in the experiment, see Fig. 4.35 for details.

4.5.5 Results

Percentages of correct sound source shape identifications are shown in Fig. 4.35 for each signal type and for front and back presentation. However, since shape 'A' (point

source) was easily identified most of the time, correct identification results for shape 'A' are not included in Fig. 4.35 so as to not bias the results.

From the subject's answers, the confusion matrices were also computed for each signal type and for frontal (Fig. 4.36) and rear (Fig. 4.37) stimulus presentation. This representation of the subject's answers is useful for highlighting the source shapes that were the most often confused. Confusion matrices also provide a quick overview of the percentages of correct sound source shape identifications when looking at the diagonal of the matrix.

F / B	Sig. Type	ETRI *[%]*	Th. *[%]*	UoW *[%]*	*Subjects*	*n*	Average *[%]*
B	white noise	16.7	33.3	21.0	*26*	*390*	**23.6**
B	low p. noise	16.0	23.7	12.4	*26*	*390*	**17.7**
B	high p. noise	20.0	30.4	17.1	*26*	*390*	**22.8**
B	blues guitar	12.7	11.1	21.9	*26*	*390*	**14.6**
F	white noise	30.0	56.3	N/A	*19*	*285*	**42.5**
F	low p. noise	16.0	35.6	N/A	*19*	*285*	**23.5**
F	high p. noise	20.0	51.9	N/A	*19*	*285*	**41.4**
F	blues guitar	12.7	N/A	N/A	*10*	*150*	**17.3**

Figure 4.35: Percentages of correct sound source shape identifications (not including shape 'A')

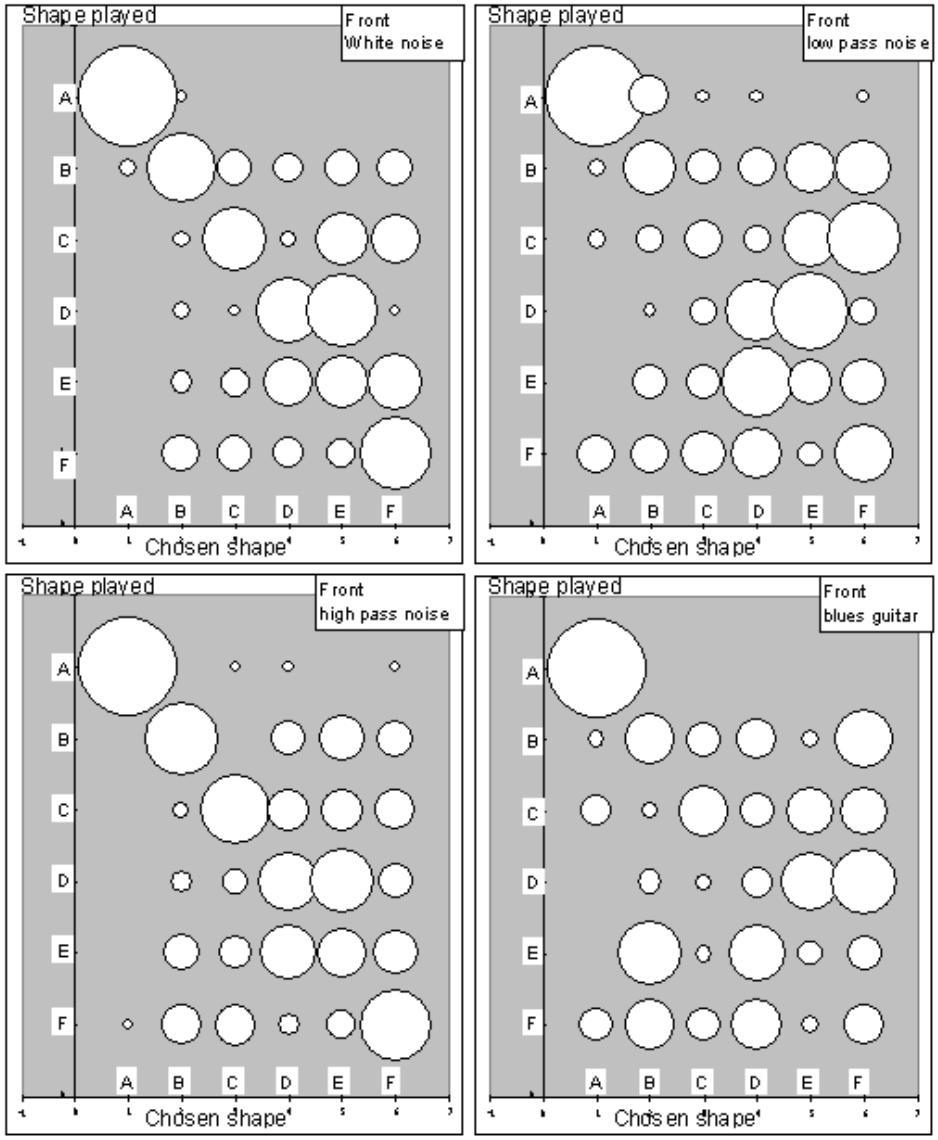


Figure 4.36: Confusion matrices of sound source shape identification for the four signal types (frontal stimulus presentation)

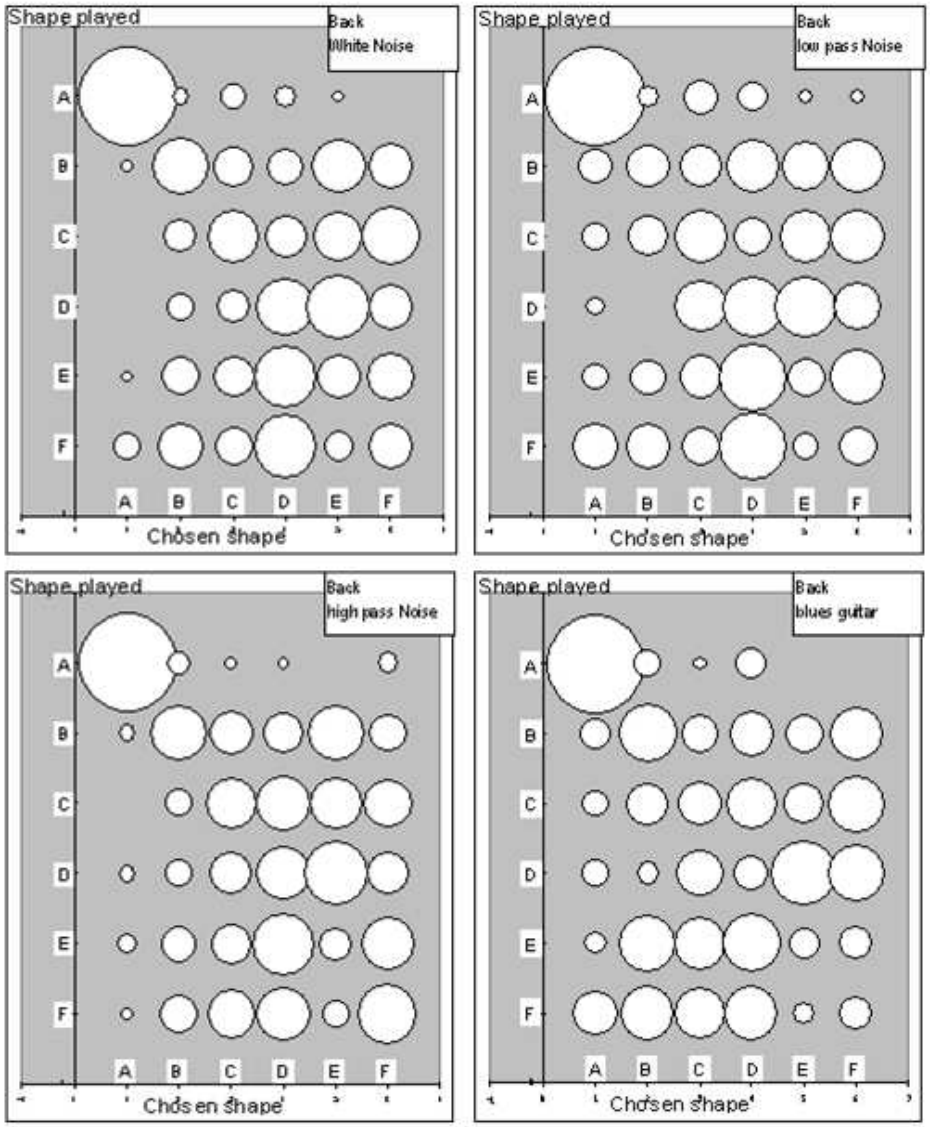


Figure 4.37: Confusion matrices of sound source shape identification for the four signal types (rear stimulus presentation)

4.5.6 Analysis of Results

Mean percentage of correct identification and confidence intervals

The average percentage of correct sound source shape identification and confidence intervals across the four signal types are plotted in Fig. 4.38 and in Fig. 4.39 for rear and frontal presentation of the stimuli. The average percentage of correct answers across the 6 sound source shapes are plotted in Fig. 4.40 and Fig. 4.41 for rear and frontal presentation of the stimuli. Fig. 4.42 shows the grand mean percentage of correct answers for front and back stimuli presentation across all shapes and signal types.

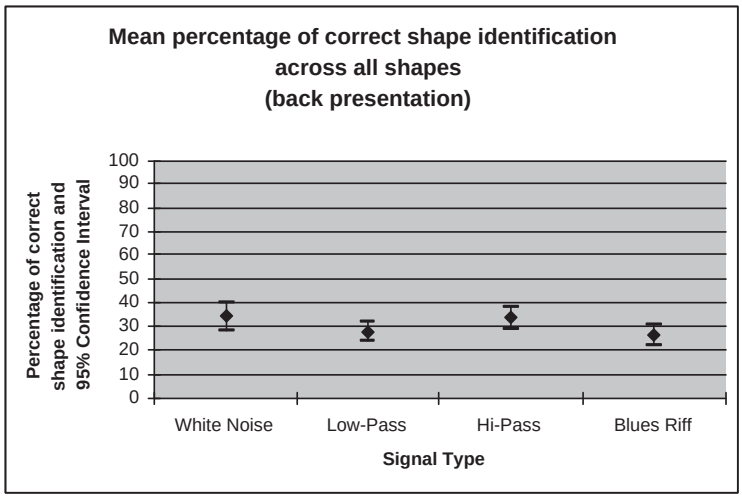


Figure 4.38: Mean percentage and 95% confidence interval of correct shape identification across four signal types, stimuli presented behind subjects

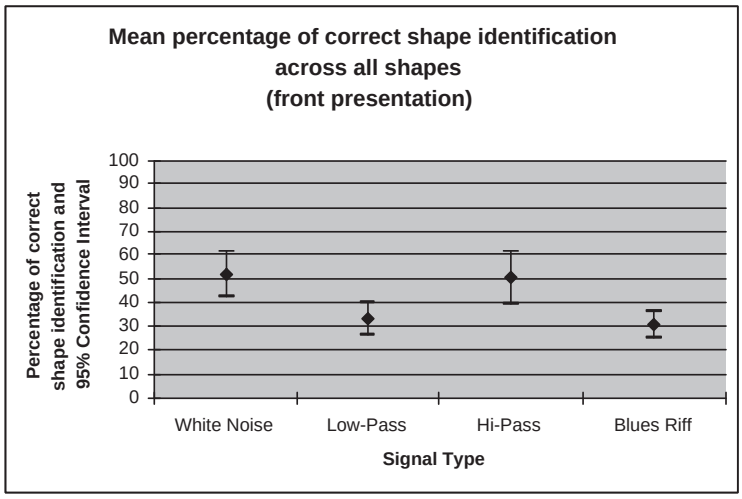


Figure 4.39: Mean percentage and 95% confidence interval of correct shape identification across four signal types, stimuli presented in front of subjects

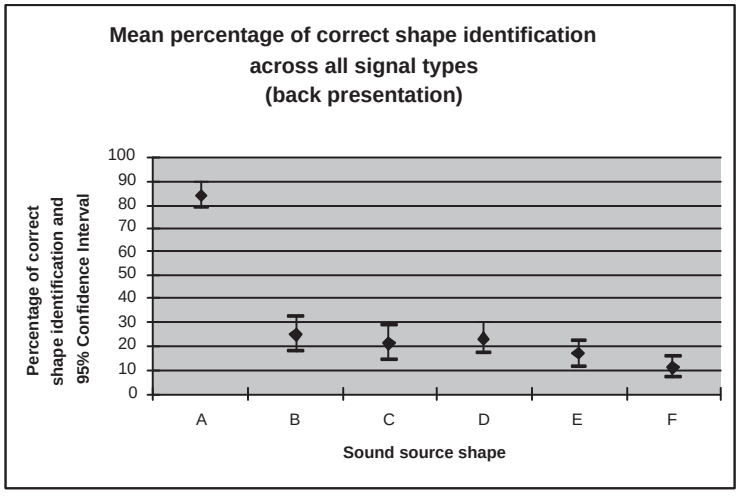


Figure 4.40: Mean percentage and 95% confidence interval of correct shape identification in function of sound source shape type, stimuli presented behind subjects (results averaged across the four signal types)

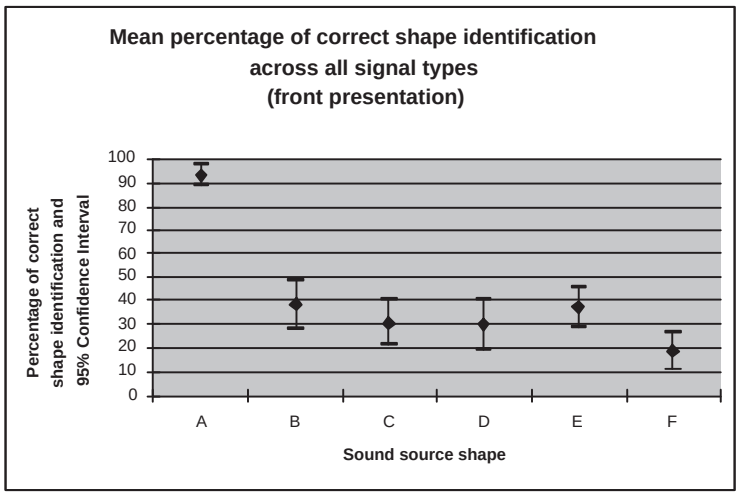


Figure 4.41: Mean percentage and 95% confidence interval of correct shape identification in function of sound source shape type, stimuli presented in front of subjects (results averaged across the four signal types)

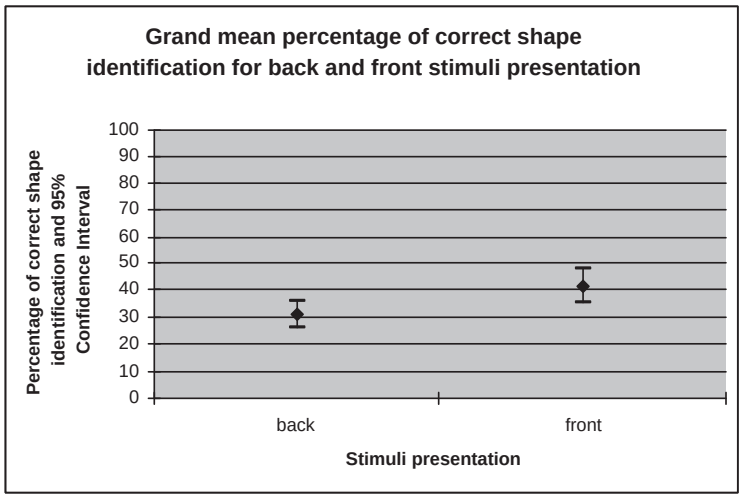


Figure 4.42: Grand mean percentage and 95% confidence interval of correct shape identification stimuli presented in the back and in front of subjects

ANOVA Analysis

A three-factor ANOVA analysis [How02] was then performed with the following factors: front/back presentation, signal type and sound source shape so as to quantify the effects of these factors on the identification of sound source shape by subjects. Results of the 3-factor ANOVA are given in Fig. 4.43. Confidence intervals smaller than 0.05, are marked with an asterisk; this indicates that the particular factor or factor interaction had a significant effect.

FACTOR	F	Sig.
FRONT/BACK	5.611	0.029 *
SIGNAL TYPE	21.635	0.000 *
SHAPE	162.684	0.000 *
FRONT/BACK * SIGNAL TYPE	2.167	0.132
FRONT/BACK * SHAPE	2.108	0.125
SIGNAL TYPE * SHAPE	6.107	0.047 *
FRONT/BACK * SIGNAL TYPE * SHAPE	0.307	0.959

Figure 4.43: 3-Factor ANOVA: F-ratios and confidence intervals

4.5.7 Discussion

Identification of sound source shape was the most successful for white noise and high-pass noise signals presented at the front of subjects (Fig. 4.35) with 42.5 % of correct identifications for white noise and 41.4 % for high-pass noise. This is well above the statistical probability (20 %)[13] but still under the 50 % threshold.

From the confusion matrices shown in Fig. 4.36 and Fig. 4.37, it can be seen that shape 'A' (point source) was easily identifiable against other sound source shapes. From the confusion matrices, it can also be seen that subjects were able to identify

[13]Five source shapes (excluding shape A), thus 1/5 = 20%

different shape categories better than the shapes themselves. These categories are :
1- Shape A (Point source)
2- Shapes B,C,D (line source)
3- Shapes E and F (2D sound source)

Reasons for this can be explained in part by the fact that perfect point sources were not used since the speakers used in the experiment had a non-zero spatial extent; therefore blurring the rendering of sound source shape. This effect can be related to the visual domain where bigger pixels on a screen decreases the picture resolution and increases image blur.

From the graphs showing the mean percentages of correct sound source shape identification for back and front presentation of the stimuli (Fig. 4.38 and Fig. 4.39 respectively), it appears that correct source shape identification was better when stimuli were presented at the front of subjects. Fig. 4.42 shows that the percentage of correct source identification was 31% for stimuli presented at the back and 41.5% for stimuli presented at the front. This finding imparts that sound source shape identification is influenced by localisation accuracy, since sound source localisation is notable for being worse at the back than at the front of listeners [Bla97]. Effects of localisation accuracy on source extent perception were also discovered in the previous experiment (section 4.4).

When comparing Fig. 4.38 and Fig. 4.39 it also appears that correct sound shape identification was better for the white noise and hi-pass signals and this trend is maintained for front or back presentation of the stimuli. This finding hints again that sound source perception is depending on source localisation accuracy as low frequency signals are notoriously hard to localise, explaining lower percentages of correct identification for the low pass noise. Regarding the blues guitar riff signal, it is possible that this signal was more difficult to localise than white noise, explaining lower identification percentages.

Another theory for explaining better shape identification for the high-pass and white noise signals is that ILD cues were used more than ITD[14] cues in the mechanisms of sound source shape perception. Indeed, ITD cues, which are mainly used at frequencies below 3kHz, are likely to have been difficult to process by the binaural system since the technique used to decorrelate signals is based on phase randomisation[15] which will tend to randomise and confuse the ITD. In this context, the binaural system is left only with ILD cues to determine the positions of the point sources and to determine sound source shape; this can explain that results were better for full-band (white noise) and high-pass noise for which ILD could be used.

Other effects such as temporal variations of the signal (as opposed to constant noise) could also have influenced source shape identification.

Figure 4.40 and 4.41 show the percentages of correct source identification for the 6 shapes for back and front presentation of the stimuli, respectively. It is apparent that shape 'A' (which was a point source) was easily identified compared to the other shapes. If shape A is removed, the statistical probability of identifying a sound source shape by chance is $1/5 = 20\%$. Thus for rear presentation of the stimuli only shape 'A' was significantly identified as for other shapes the confidence interval overlaps with the 20% statistical chance limit. For frontal presentation of the stimuli, only shape 'F' was not significantly identified above statistical chance.

From results of the ANOVA analysis shown in Fig. 4.43 it can be seen that front or back presentation of the stimuli significantly affected the ability by subjects to identify sound source shapes. Signal type was also a significant factor with a relatively high F-ratio. As expected, the shape factor highly affected correct shape identification, in particular as it was shown in Fig.4.38 and Fig.4.39 that shape 'A' was very easily identified against the other sound source shapes. Factor interactions were mostly insignificant except for (signal type * shape) albeit with a relatively low F-ratio. This significant interaction can be related to localisation of the point sources forming the

[14]ILD and ITD cues were reviewed in section 2.5.1

[15]This was explained in section 2.13

shapes which depended on the type of signal emitted by the sound sources (and was worse for the low-pass noise and blues guitar riff signals).

Results of the ANOVA analysis formally confirm the trends that were previously discovered.

Results of the experiment showed that correct sound source shape identification was dependent on sound source localisation accuracy and this resulted in poor shape identification for stimuli presented behind subjects and for certain types of signal (low pass noise and blues guitar riff) which were harder to localise. This also raises concerns that the rendering of sound source shape using traditional decorrelation techniques collapses for certain types of signals, especially signals with time-varying intensity and spectrum (such as music for instance). The point sources used to represent the sound source shapes were not actually punctual as each source was a speaker, this again might have blurred the perception of sound source shape.

The use of real speakers to represent the shapes is impractical in a real-case scenario as a large number of speakers and channels are required. The topic of the next experiment is to study the identification of sound source shapes where the decorrelated point sources are not actual physical sound sources but spatialised virtual sound sources.

4.6 Experiment 4: perception of sound source shape using virtual decorrelated sound sources

4.6.1 Aims

The aim of this experiment is to study the perception of sound source shapes that are created using virtual decorrelated point sources; to do so, the decorrelated point

source technique (section 2.11) is combined with the Ambisonics spatialisation technique (section 5.4.1). This experiment corresponds to a more real scenario for producing sound source shapes in 3D auditory systems compared to the apparatus of the previous experiment (section 4.5). This experiment also studies the effects of decorrelation on sound source shape perception by comparing the perception of sound source shapes devised with decorrelated and correlated sound sources. It is expected that shape perception abilities will be reduced when using correlated point sources due to the summing localisation effect which was explained in section 2.12.2.

This study was first conducted for the MPEG Audio group to study the need of implementing sound source shape capabilities in the MPEG-4 standard (see section 4.10 for details). The experiment was designed by the author and was carried out at two locations: University of Wollongong and Thomson Multimedia (Germany). The participation of Jens Spille[16] for repeating this experiment at his laboratory is acknowledged.

4.6.2 Apparatus

The apparatus consisted of an 8-speaker array arranged in a cubic configuration (Fig. 4.44). The cube had dimensions of approximately 1.6 x 1.6 x 1.6 meters. High-quality Genelec 1029A speakers equalised four loudness were used.

4.6.3 Stimuli

Five different sound source shapes were created using a constant number of four point sources; these are depicted in Fig. 4.45. The angular extent of the sound source shapes was 80 degrees horizontally and 90 degrees vertically; coordinates of the spatialised point source are detailed in Fig. 4.46. First order Ambisonics spatialisation[17] was used to place the virtual sound sources on the cubic speaker array. All sound source

[16]Research Engineering at Thomson Multimedia, Germany
[17]see section 5.4.1

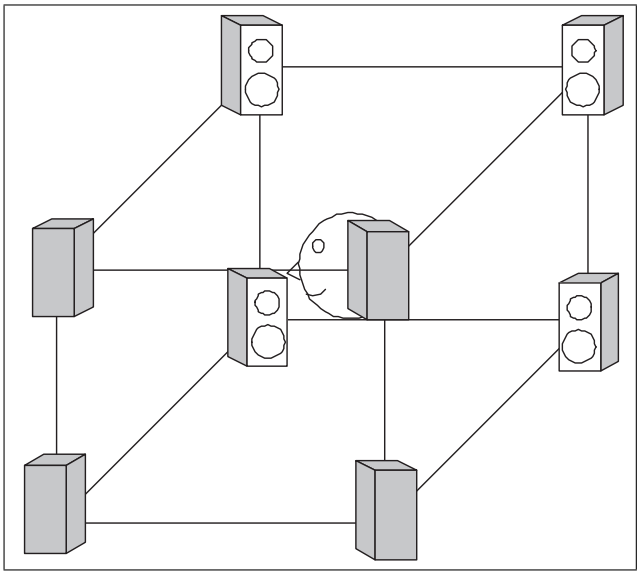


Figure 4.44: Placement of subjects at the centre of the speaker cube apparatus

shapes appeared in front of subjects so as to maximise localisation precision [Bla97].

To create the stimulus sequences, Ambisonics B-format (section 5.4.1) files were created; these contained the spatialised point sources that formed the sound source shapes. In a first time, ten sequences were created where the five shapes depicted in Fig. 4.45 were played twice; the signals emitted by the point sources were uncorrelated white noise sequences. In a second time, ten further sequences were obtained where the signals emitted by the sound sources were identical (i.e. correlated) white noise sequences. To perform the experiment, the created B-format sequences were decoded for the cubic speaker array apparatus shown in Fig. 4.44. Details for performing such decoding can be found in [FM].

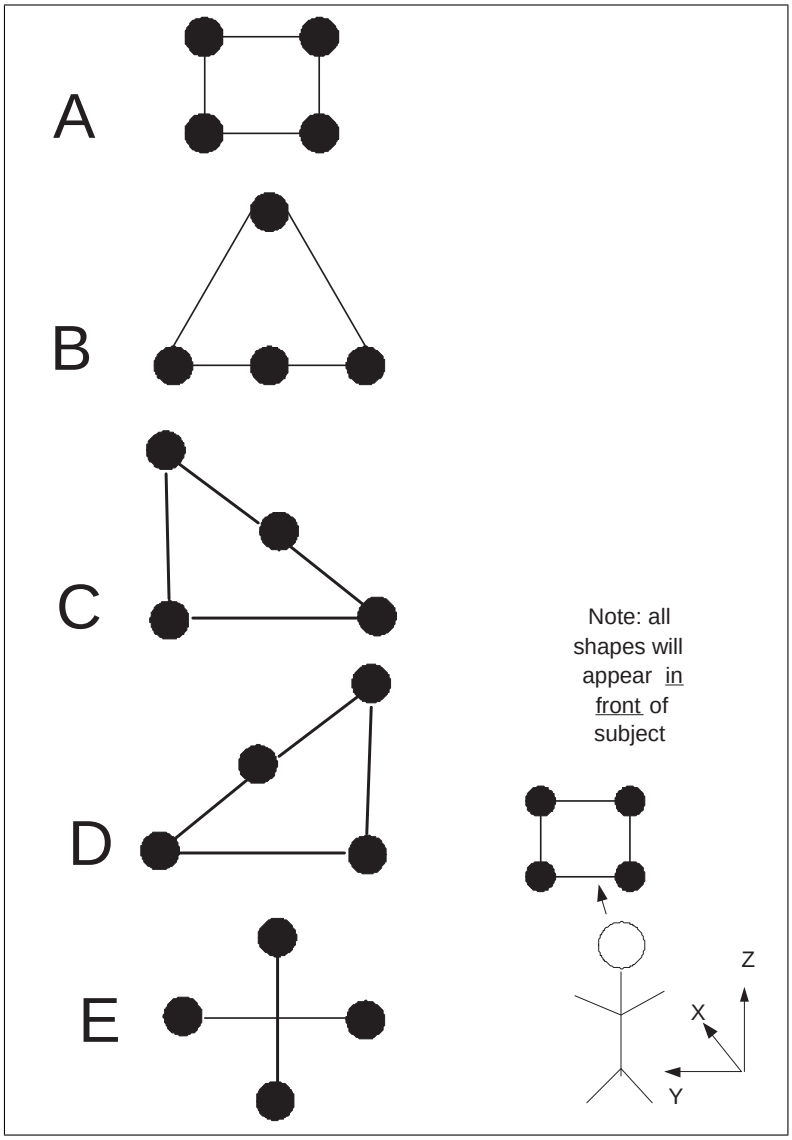


Figure 4.45: Geometry of the five sound source shapes

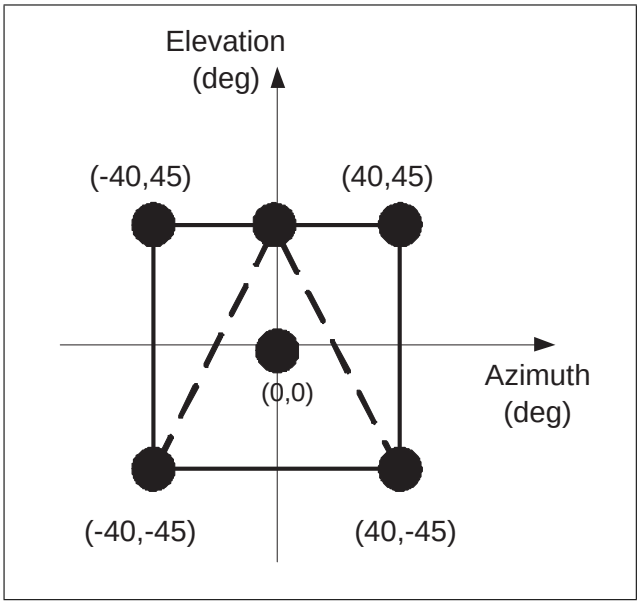


Figure 4.46: Coordinates of the point sources used to form the sound source shapes

4.6.4 Procedure

For each of the twenty stimuli, which were played in random order, subjects were asked to identify an apparent source shape among five possible shapes. Subjects were allowed to leave a blank answer if they could not identify the shape of a particular sequence. Head rotations were allowed and subjects were not visually masked. Before starting the experiment, training was given to subjects so that they listened to each of the five sound source shapes. In all, sixteen adult subjects with normal hearing took part in the experiment (seven at the University of Wollongong and nine at Thomson).

4.6.5 Results

Confusion matrices for decorrelated and correlated point sources are shown in Fig. 4.47 and Fig. 4.48 respectively.

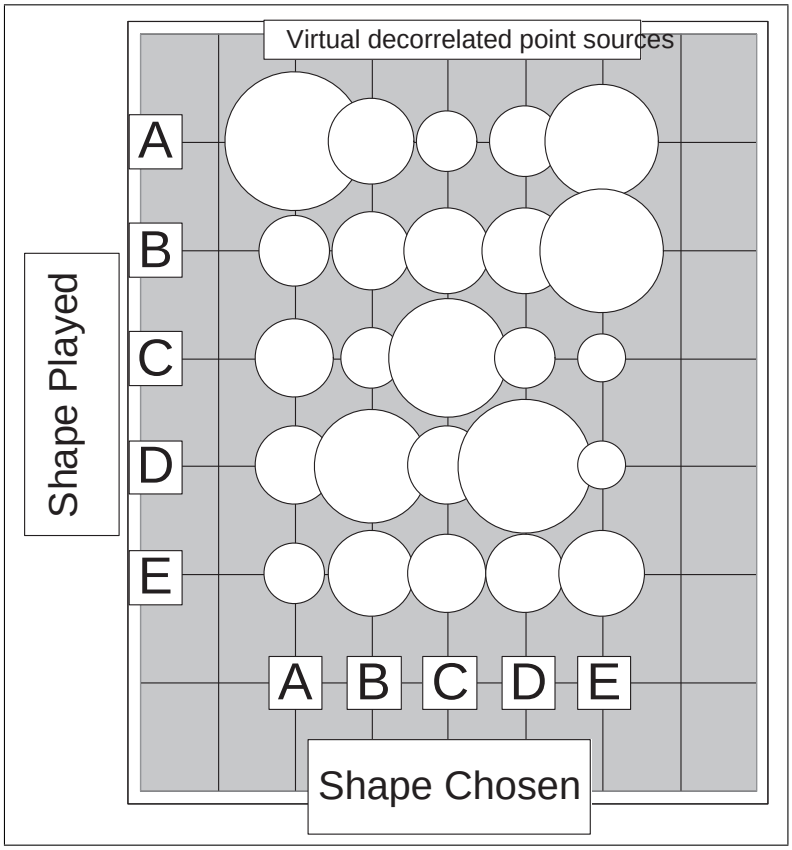


Figure 4.47: Confusion matrix of shape identifications (shapes created with decorrelated virtual sound sources)

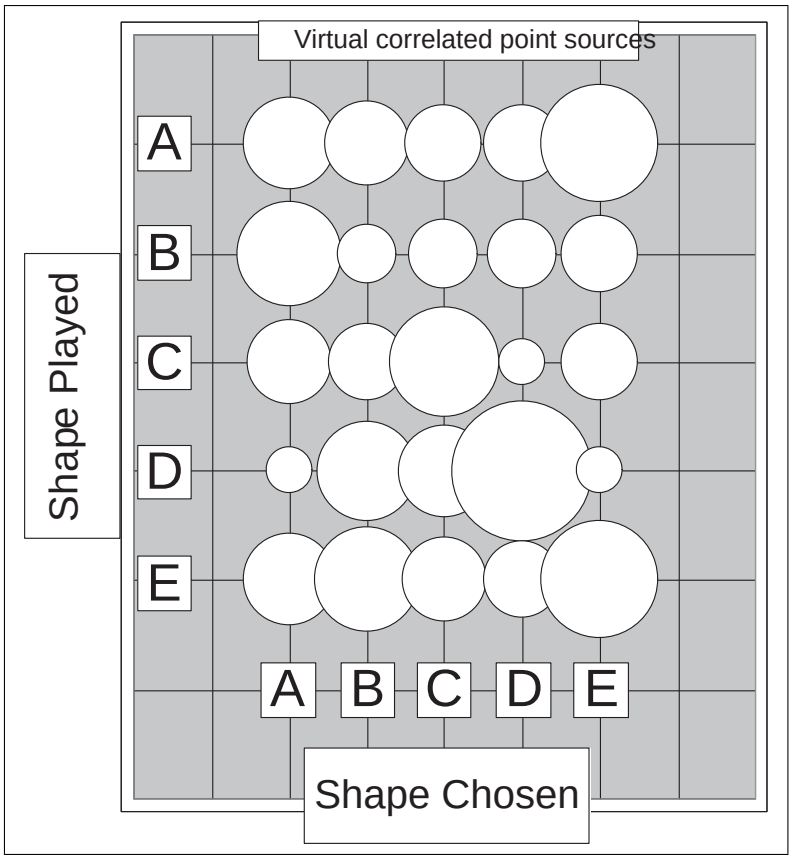


Figure 4.48: Confusion matrix of shape identifications (shapes created with correlated virtual sound sources)

4.6.6 Analysis of results

Mean percentage of correct identification and confidence intervals

The grand mean percentage of correct sound source shape identification and confidence intervals for decorrelated and correlated point sound sources are shown in Fig. 4.49. The average percentages of correct sound source shape identification and confidence intervals for the five sound source shapes are plotted in Fig. 4.50 and Fig. 4.51 for decorrelated and correlated point sources respectively.

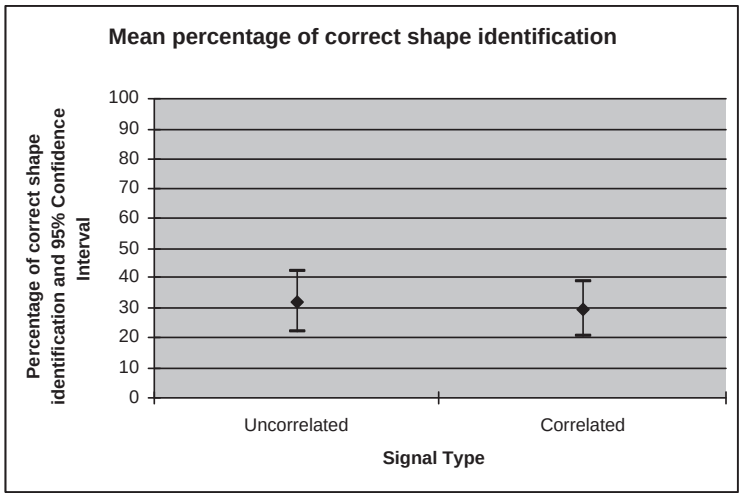


Figure 4.49: Mean percentage and 95% confidence interval of correct shape identification for decorrelated and correlated point sound sources

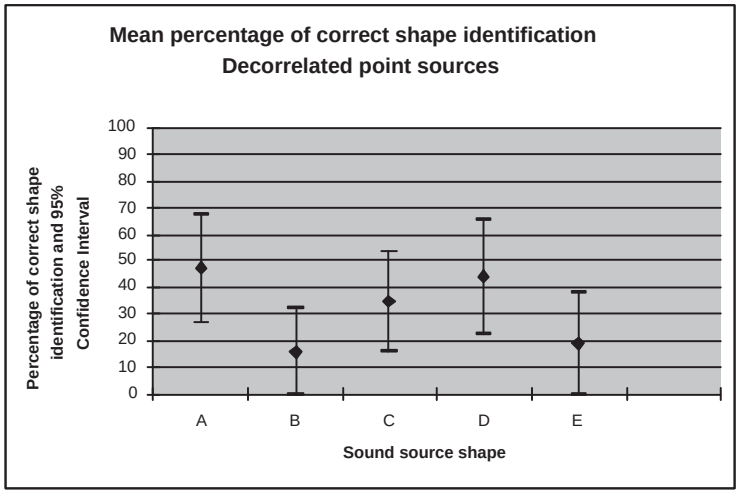


Figure 4.50: Mean percentage and 95% confidence interval of correct shape identification for the five sound source shapes for decorrelated point sound sources)

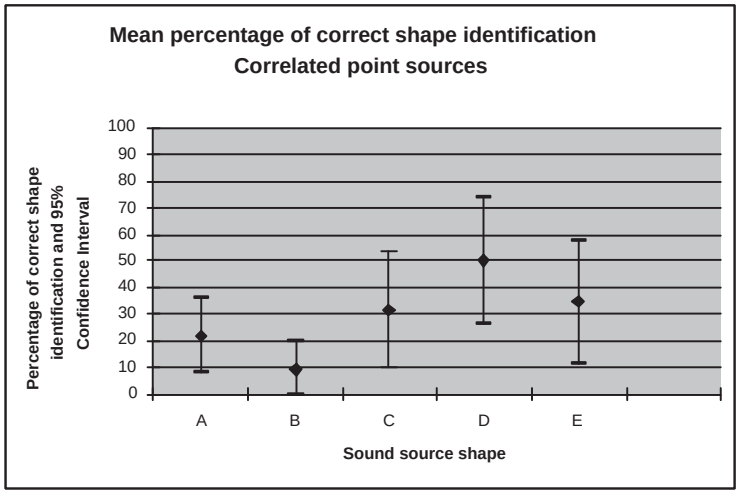


Figure 4.51: Mean percentage and 95% confidence interval of correct shape identification for the five sound source shapes for correlated point sound sources)

ANOVA analysis

A two-way ANOVA analysis was performed with the following factors: decorrelated or correlated point sound sources, and sound source shape. Results of the two-way ANOVA are given in Fig. 4.52. Confidence intervals smaller than 0.05, are marked with an asterisk; this indicates that the particular factor or factor interaction had a significant effect.

FACTOR	F	Sig.
Uncorrelated/correlated	0.306	0.588
Shape	6.6	0.005 *
Uncorrelated/correlated * Shape	0.819	0.537

Figure 4.52: 2-Factor ANOVA: F-ratios and confidence intervals

4.6.7 Discussion

As shown in Fig .4.49, the difference of percentage of correct shape indentification between decorrelated and correlated point sources is not significant. In both cases, however, sound source shapes were, in average, identified just above statistical chance (5 shapes thus 20% probability).

Fig. 4.50 shows the percentages and 95% confidence intervals of correct shape identification for the five different shapes used in the stimuli. It appears that only shape 'A' (square) and 'D' (triangle pointing to the left) were identified significantly above statistical chance. For correlated sound sources (Fig. 4.51), only shape 'D' was identified above statistical chance. It remains unexplained why shape 'C' which was a triangle pointing to the right was not identified as often as shape 'D'.

It is interesting to note that the percentages of correct identification for each shape follow the same pattern for decorrelated and correlated point sources. This indicates

that other cues may have been used to identify the sound source shapes, such as localisation of the centre of gravity of the point sources or colouration of the white noise used in the stimuli.

The confusion matrices for decorrelated and correlated point sources are shown in Fig. 4.47 and Fig. 4.48 respectively. From these, it can be seen that shapes 'A' and 'D' were the most often successfully identified and shape 'B' was the least often successfully identified shape. As far as shape 'E' is concerned, its poor identification by subjects can be explained by the observation that shape 'E' could be perceived either as a cross or a diamond, and that shape 'E' was more easily confused with shape 'A', since the two shapes are geometrically close. This is shown by a relatively high value between 'A' and 'E' in in the confusion matrices.

The ANOVA table shown in Fig. 4.52 confirms that the decorrelation or not of the point sound sources was not a significant factor in the correct identification of the sound source shapes (this was also shown in Fig .4.49); this finding was unexpected. On the other hand, geometry of the sound source shapes is a significant factor and percentage of correct identification varies for each shape; this was discovered in Fig. 4.50 and Fig. 4.51. The ANOVA analysis also indicates that there was no interaction between the decorrelation/correlation factor and the shape of the sound sources.

The rendering device used to produce the stimuli consisted of 8 speakers arranged in a cube and 1st order Ambisonics spatialisation was employed. This configuration is the bare minimum for rendering periphonic 3D audio and due to the wide separation of speakers this results in highly unstable spatialised sound sources (during head motions) when these are located in between speakers. Thus the limitation of the speakers is likely to be the main cause for poor identification of the sound source shapes (although some shapes were significantly identified above pure chance).

The limitation of the rendering system also explains that decorrelation or not of the point sources did not significantly affect shape identification.

In conclusion, this experiment showed that the rendering of sound sources shapes

using virtual sound sources requires an accurate 3D audio spatialisation technique, unlike the apparatus used in the experiment. The identification of certain shapes ('A' and 'D') above statistical chance may have been helped by extra cues such as localisation of the centre of gravity and noise colouration.

4.7 Experiment 5: improvement in 3D audio scene realism by using extended sound sources

Having studied the perception of rendered sound source extent and shape in 3D audio displays, the perceptual impact of using extended sound sources in 3D audio scenes is now studied. This experiment aims at studying the necessity of rendering sound source extent in 3D audio scenes.

This study was first conducted for the MPEG Audio group to study the need of implementing sound source extent capabilities in the MPEG-4 standard (see section 4.10 for details). The experiment was created by the author and was carried out at two locations: University of Wollongong and ETRI (Korea). The participation of Jeongil Seo[18] for repeating this experiment at his laboratory is acknowledged.

4.7.1 Aims

This experiment studies the perceived *naturalness* of 3D audio scenes that incorporate broad sound sources over 3D audio scenes that use only point sources. To do so, subjects were asked to perform A-B comparisons between these two types of 3D audio scenes. The sound sources present in the test scenes referred to auditory events that are normally perceived as being broad and spacious (crowd, thunder, truck, beach, city and water). The experiment was first carried out on speakers then repeated

[18]Research Engineering at ETRI, Korea

on headphones, using binaural spatialisation. This was in order to verify the improvement in naturalness when using broad sound sources in the context of these two spatialisation techniques.

4.7.2 Apparatus

The apparatus consisted of a 7-speaker array which was the same as used in the experiment described in section 4.5; this apparatus was depicted in Fig. 4.31. Binauralised audio scenes were rendered on headphones.

4.7.3 Stimuli

Speaker stimuli

Six different sound recordings that referred to naturally large auditory events or objects (crowd, thunder, truck, beach, city and water) were used to create 3D audio scenes that were either employing these objects as point sound sources (one speaker of the array used) or as broad sound sources (seven speakers of the array used, emitting decorrelated signals). To obtain seven decorrelated signal replicas of the different sound recordings, a 256-tap FIR decorrelation filterbank[19] was used. Comparison sequences were then created whereby a first scene was played followed by a second scene. The order of appearance of the scenes (point source or spatially extended) was randomised. In all, subjects had to compare six pairs of 3D audio scenes for the six types of signal (crowd, thunder, truck, beach, city and water).

Headphones stimuli

The sequences created for speaker playback were also binaurally encoded at the positions of the speakers of the array. To do so, a dummy head HRTF database was used to binaurally spatialise the point sources of the speaker array. This was done

[19]See section 2.13.2

in Matlab for each speaker by performing binaural convolution of the speaker signal with the HTRF filter measured at the azimuth and elevation of the speaker. Binaural signals obtained for the seven speakers were then added together to form the binaurally encoded stimuli.

4.7.4 Procedure

For each sequence, subjects were asked to perform A-B comparisons in terms of *naturalness* between the first and second played 3D audio scene. Eighteen adult subjects with normal hearing participated in the experiment on speakers and six participated in the experiment on headphones.

4.7.5 Results

Percentages of time where 3D audio scenes that used spatially extended sound sources were preferred (in terms of naturalness) over scenes that used only point sources are shown in Fig. 4.53 (figure shows results obtained at ETRI and University of Wollongong).

	ETRI	UoW	*Total Participants*	*Average*
Speak.	77.5%	68.70%	18	70.40%
Head.	N/A	69.40%	6	69.40%

Figure 4.53: Percentages of time where 3D audio scenes that used extended sound sources were subjectively preferred (for speaker and headphone stimulus presentation)

4.7.6 Discussion

Fig. 4.53 shows that 3D audio scenes that used spatially extended sound sources were perceived as being more natural 70.4 % of the time on speakers and 69.4 % on headphones. This experiment thus showed that the rendering of sound source extent in 3D auditory is perceptually relevant and that better aesthetics of the rendered 3D audio scenes are achieved. Indeed, the signals used in the stimuli referred to large auditory events, and thus representing them using point sources sounded unnatural. These results corroborate the knowledge that spacious sound fields are generally more pleasing that non-spacious ones (in concert halls) [Gri97].

4.8 Experiment 6: perceptual effects of dynamic decorrelation

4.8.1 Aims

This experiment investigates the benefits of using dynamic decorrelation in order to provide more realism in 3D audio scenes. The listening fatigue caused by dynamic decorrelation is also studied. Dynamic decorrelation techniques were reviewed in section 2.13.6.

4.8.2 Apparatus

The experiment was carried out using good quality headphones.

4.8.3 Stimuli

Stimuli were produced from the *monaural* recording of a noisy market in Nice, France that contained many sound sources (sellers yelling, cars, children etc.). From this

monaural recording, three pseudo-stereo[20] versions of the recording were obtained. This was achieved by using a 128-order IIR decorrelation filterbank (section 2.13.3) with fixed and time-varying coefficients (with coefficient update windows of 1s and 100ms). This way, fixed and dynamically (at 1Hz and 10Hz) decorrelated signals were obtained. The obtained pseudo-stereo recordings were then arranged in sequences, containing a first pseudo-stereo recording followed by two seconds of silence and followed by the second pseudo-stereo recording. Three sequences were formed in this way:

1- Fixed decorrelation – Dynamic decorrelation (100ms)

2- Dynamic decorrelation (100ms) – Dynamic decorrelation (1s)

3- Dynamic decorrelation (1s) – Fixed decorrelation

4.8.4 Procedure

Subjects had to perform A-B comparisons in terms of *naturalness* for each of the three created sequence. To study the listening fatigue caused by dynamic decorrelation, subject also rated the fatigue of the original monaural recording and the three pseudo-stereo recordings on a 1 to 5 scale, 1 corresponding to no fatigue and 5 to extreme fatigue. Eight subjects participated in the experiment.

4.8.5 Results

Table 4.1 shows the percentages of preference in terms of naturalness between the three types of decorrelation. Table 4.2 shows the average fatigue for no signal decorrelation and for the three types of decorrelation (fixed and dynamic decorrelation with 100ms and 1s update rates).

[20]Pseudo-stereo was reviewed in section 2.11.2

Comparison	% first	% same	% second
Fixed-Dynamic 100ms	40	0	60
Dynamic 100ms-Dynamic 1s	20	33.3	46.7
Dynamic 1s -Fixed	60	6.6	33

Table 4.1: Percentages of preference in terms of naturalness between fixed and dynamic decorrelation

Type of decorrelation	average fatigue (1-5)
Fixed	1.625
Dynamic 100 ms	2.625
Dynamic 1 s	2.5
None	1.5

Table 4.2: Average listening fatigue caused by fixed, dynamic and no decorrelation (1: no fatigue, 5: extreme fatigue)

4.8.6 Discussion

Results in table 4.1 show that dynamic decorrelation was perceived as being more natural than fixed decorrelation. This can be explain as follows: dynamic decorrelation created an interesting effect whereby the sound sources of the original recording were moved around the scene (these movements were not present in the original monaural recording). This effect, which is caused by dynamic decorrelation was explained in section 2.13.6. The sound recording which was used (market) implied movement and a busy activity around the listener; therefore this can explain that subjects preferred dynamic decorrelation over fixed decorrelation since the latter did not produce movements. However, slower (1s) dynamic decorrelation was preferred over faster (100ms) dynamic decorrelation because fast decorrelation created too fast movements of objects in the scene, and this was unnatural. Therefore, from this experiment it can be concluded that the use of dynamic decorrelation is beneficial to the realism of 3D audio scenes however its use depends on the nature of the signal to decorrelate.

Table 4.2 shows the average fatigue for each type of decorrelation and no signal decorrelation. The scale ranged from 1 (no fatigue) to 5 for (extreme fatigue). As expected, dynamic decorrelation created more fatigue than fixed decorrelation due to movements of objects in the scene it produced, and fatigue was higher for faster decorrelation. The original recording (no decorrelation) was perceived as the least fatiguing.

In conclusion, dynamic decorrelation can be used to improve the naturalness of 3D audio scenes, however it must be used carefully so as to not create too much listening fatigue. The nature of the signal to decorrelate should also be taken into consideration when choosing the update rate of the dynamic decorrelation filters.

4.9 Experiment 7: perceptual effects of time-varying decorrelation

4.9.1 Aims

This experiment aims at finding, by using time-varying decorrelation, the time constant of the binaural system at which it is able to perceive changes in the inter-aural cross-correlation coefficient (IACC)[21].

4.9.2 Apparatus

The experiment was carried out using good quality headphones.

4.9.3 Stimuli

To study the sensitivity of listeners to time varying decorrelation, two white noise signals which had a periodic change of correlation between them were used. To do

[21]Defined in 2.8.2

so, the time varying decorrelation technique was used; this technique was reviewed in section 2.13.8. The temporal variations of the cross-correlation coefficient between the two signals followed a sine wave pattern which frequency could be controlled in real time. The range of variations of the cross-correlation coefficient between the two signals was 0 (decorrelated signals) to +1 (identical signals). Since the two signals were presented on headphones, the cross-correlation coefficient between them directly drove the inter-aural cross correlation coefficient (IACC).

A value of +1 (or close to 1) of the cross-correlation coefficient between the two signals (and hence of the IACC) produced a narrow sound image located within the head. On the other hand, a value of 0 (or close to 0) of the cross-correlation coefficient between the two signals produced a wide sound image, and the two noise signals were localised at the subject's ears. Effects of the IACC coefficient on the image width of noise presented on headphones was reviewed in section 2.8.4. Therefore, the effect of periodically varying the level of decorrelation between the two white noise signals produced a sound image that periodically varied from a narrow central image to a spacious, almost external image.

4.9.4 Procedure

The experiment consisted of increasing the rate of change of the cross-correlation coefficient between the two signals until the subject indicated that he/she could not longer perceive variations of the sound image width. The value of the rate of change frequency was then recorded. The reverse experiment was also repeated in which a high frequency rate of change of the cross-correlation coefficient was decreased until the subject indicated that he/she perceived variations of the sound image width; the value of this frequency was again recorded. Eight subjects took part in the experiment.

4.9.5 Results

The averages of the rate of change frequencies at which subjects perceived no more variations in sound image width (for raising and decreasing frequency of the rate of change of the IACC coefficient) are shown in table 4.3 and plotted in Fig. 4.54. The corresponding time windows (i.e. 1/f) of the average rate of change frequencies are also shown.

Variation of the rate of change	Rate of change frequency	Time window
Low $\longrightarrow$ High	12.5 Hz	80ms
High $\longrightarrow$ Low	11 Hz	91ms

Table 4.3: Average frequencies of the rate of change of the IACC at which subjects could no more perceive a change in source extent

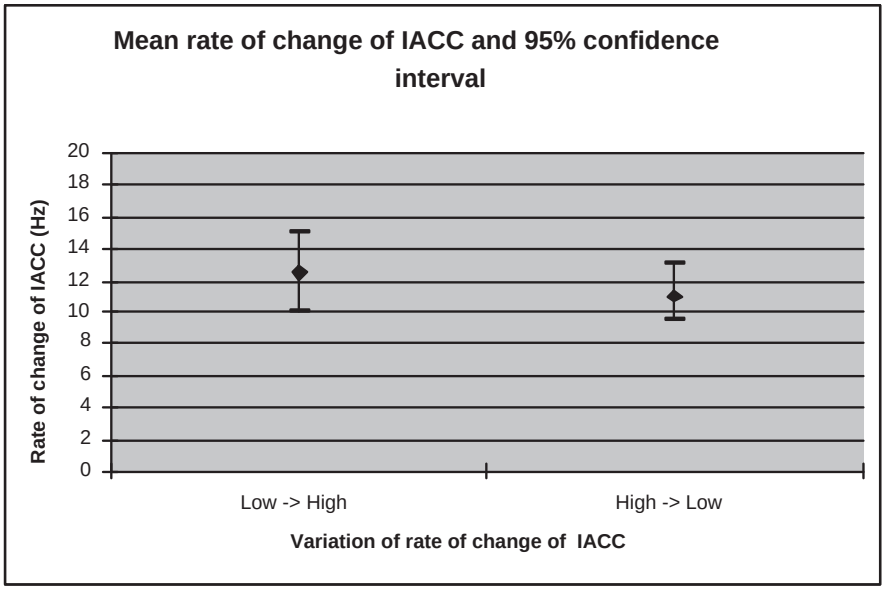


Figure 4.54: Mean rate of change of IACC and 95% confidence interval at which subject perceived no more change in sound image width

4.9.6 Discussion

This experiment allowed to derive an average value for the time constant of the binaural system at which it is able to perceive changes in IACC. Figure 4.54 shows that there was no significant change in this time constant whether the rate of change of the IACC was increasing or decreasing, as the confidence intervals are overlapping.

It was decided to take the average of the two rate of change frequencies for the final value (85.5 ms) at which the binaural system computes a new IACC coefficient. This result is comparable to that of the study of Chait et al. [CPCS05] who found a value of 80 ms.

Faster changes in inter-aural correlation are thus not perceptible. This finding has applications in using the time-varying decorrelation method described in 2.13.8 and has applications in acoustical engineering where it was shown that temporal fluctuations of the IACC influence the perception of spaciousness (see section 2.9).

4.10 Implementation of sound source extent description capabilities in MPEG-4 AudioBIFS

The implementation of sound source extent capabilities in MPEG-4 AudioBIFS[22] Version 3 is now reviewed. After an initial document [PB02a] by the author which highlighted the lack of sound source extent description capabilities in MPEG-4 AudioBIFS and which proposed a technique to add this feature in the MPEG-4 standard, three MPEG Core Experiments (CE) designed by the author were carried out[23]. These experiments, described in section 4.3, 4.5, 4.6, 4.7 and 4.8 were performed with a view to study the need and feasibility of implementing sound source extent and shape in the MPEG-4 AudioBIFS standard. Results of the experiments were presented at three consecutive MPEG conferences: Shanghai, October 2002 [PS02]; Awaji Island (Japan), December 2002 [PSS02]; Pattaya (Thailand), March 2003 [PSS03].

The first core experiments [PS02, PSS02, PS03] studied the usefulness of implementing sound source shape description capabilities in AudioBIFS. However, due to results showing that subjects were able to identify sound source shapes only less than 50 % of the time[24], it was decided that an *exact* sound source shape description capability in AudioBIFS was overkill.

However, since it was found in a last core experiment [PSS03] that subjects were accurate in determining the angular extent of sound sources (section 4.3), and that it was found that listeners could identify one-dimensional and two-dimensional sound sources (section 4.4), it was decided that sound source extent would be described in MPEG-4 AudioBIFS using primitive geometrical shapes, that is: lines, rectangles, rectangular boxes, cylinders and spheres; these are shown in Fig. 4.55. These shapes can be further oriented in the 3D audio scenes and X, Y and Z source dimensions can be specified to create 1D, 2D and 3D extended sound sources. A special extent can

[22]MPEG-4 AudioBIFS was reviewed in 2.4.2

[23]Experiments were carried out by the author and repeated by Jens Spille of Thomson Multimedia (Germany) and Jeongil Seo of ETRI (Korea)

[24]See experiments described in section 4.5 and 4.6

also be specified (shape 1 in Fig. 4.55); this case allows describing a source extent independently of the listening/view point. This is useful when AudioBIFS scene authors require a constant source extent that is independent of the listening/view point in the scene, such as, for instance, to create spatially wide sound atmospheres (e.g. wind in trees, thunder etc.).

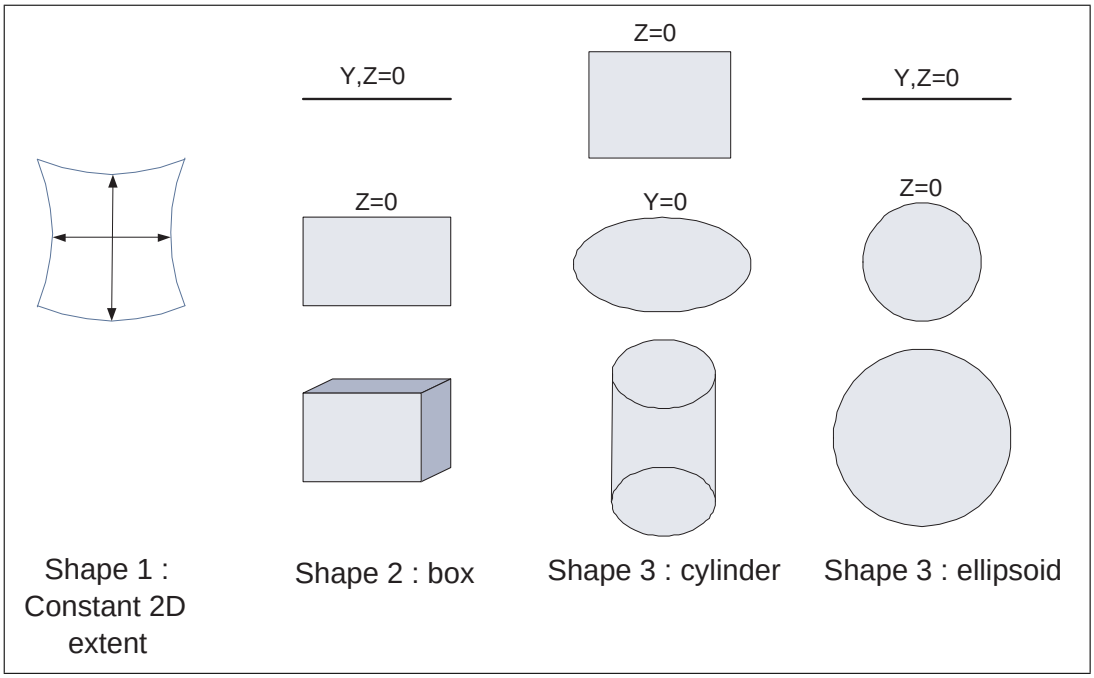


Figure 4.55: Different sound source shape types definable in the field of the WideSound node

To provide such sound source extent description capabilities in MPEG-4, a new AudioBIFS node called *WideSound* was created. This new node is included in the current MPEG-4 AudioBIFS V3 working document which will reach Final Draft International Standard (FDIS) status in 2005. The semantics of the new *WideSound* node are shown in Fig. 4.56. The *shape* field is used to select the shape of the sound source as depicted in Fig. 4.55. The *size* field is used to specify the dimensions of the sound source in the X, Y and Z planes; this permits creating sound sources with one, two and three-dimensional extents (Fig. 4.55). The *density* field specifies the

density[25] of the point sources and the *decorrStrength* field specifies the inter sound source correlation coefficients. The *diffuseSelect* field specifies a flag that indicates the scene renderer to use different decorrelation filters so that if the same audio stream is used by several *WideSound* nodes, the correlation coefficients between the sound sources remain zero. Other fields of the *WideSound* node are identical to that of the *DirectiveSound* node of AudioBIFS version 2 [MPE01].

WideSound {

	exposedField	SFFloat	**intensity**	1	
	exposedField	SFVec3f	**location**	0, 0, 0	
	exposedField	SFNode	**source**	NULL	
	exposedField	SFBool	**spatialize**	TRUE	
	Field	SFFloat	**speedOfSound**	340	
	Field	SFFloat	**distance**	1000	
	Field	SFBool	**useAirabs**	FALSE	
	exposedField	SFNode	**perceptualParameters**	NULL	
	exposedField	SFBool	**roomEffect**	FALSE	
	exposedField	SFInt32	**shape**	0	
	exposedField	MFFloat	**size**	0	
	exposedField	SFVec3f	**direction**	0, 1, 0	
	exposedField	SFFloat	**density**	0.5	
	exposedField	SFInt32	**diffuseSelect**	1	
	exposedField	SFFloat	**decorrStrength**	1	0..1

}

Figure 4.56: Semantics of the new *WideSound* AudioBIFS node to represent sound sources with apparent extents in MPEG-4 AudioBIFS scenes

An example of the use of *WideSound* nodes in a MPEG-4 AudioBIFS 3D audio scene is shown in Fig. 4.57. The example 3D audio scene consists of listening to a choir in an auditorium, from the listening point of a spectator located in the middle of the audience. After the performance of the choir, the audience applauds. This scene can be easily composed using four *WideSound* nodes and audio streams/files for the choir recording and applause sound. In AudioBIFS version 2, such 3D audio scene could not have been devised and point sources had to be used instead; resulting in poor naturalness, spaciousness and listener immersion.

[25]Effects of point source density were studied in section 4.3

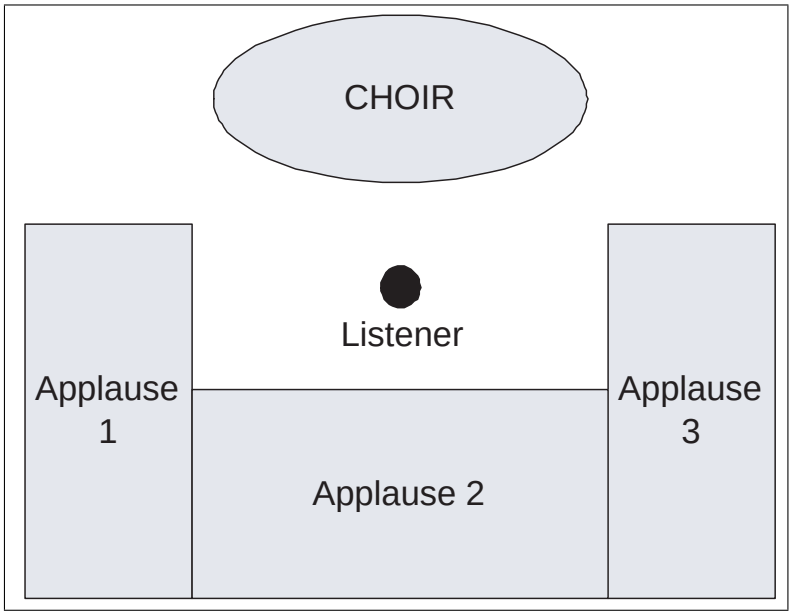


Figure 4.57: MPEG-4 AudioBIFS 3D audio scene example containing four *WideSound* nodes

4.11 Summary

This chapter presented a novel psychoacoustic study which explored the ability by listeners to perceive the apparent extent and shapes of sound sources. The stimuli were created using the decorrelated sound source technique which was first reviewed in section 2.12. A new hypothesis was proposed which stated that, by arranging the positions of the decorrelated sound sources into particular patterns, this technique could be used to create sound sources with certain apparent shapes.

Two experiments presented in section 4.5 and 4.6 aimed at validating this hypothesis by studying the ability of subjects to identify apparent sound source shapes. The shape perception experiment was, in a first time, carried out with real decorrelated sound sources as to provide maximal stability and localisation precision. When white noise was used and when the source shapes were presented in front of subjects, subjects could identify the source shapes 42.5 % of the time (statistical chance was

20 %). However, when a music signal was used or when the shapes were presented in the back of subjects, shape identification was equal or less than statistical chance. In experiment 4 (section 4.6), the decorrelated sound sources where positioned using spatialisation. As expected, shape identification was reduced (31.9 % of correct identifications with white noise) due to errors introduced by spatialisation.

In experiment 1 (section 4.3), the effects of density of the decorrelated sound sources on perceived horizontal extent were studied. It was found that, while loudness and signal type did not have much impact on perceived horizontal extent, density had the most impact. Excessive density resulted in a narrower perceived extent, on the other hand, insufficient density resulted in loss of binaural fusion (i.e. the different decorrelated sound sources were distinctly perceived). Density of the decorrelated sound sources is thus a major parameter when rendering source extent in 3D audio displays and should be careful controlled.

In experiments 2 and 3, described in sections 4.4 and 4.5, effects of sound localisation precision were studied. It was shown that the precision at which subjects could perceive the apparent extents and shapes of the stimuli was decreased when the stimuli where presented at position at which sound localisation is worse (i.e. on the sides, at the back and above subjects). Thus, the perception of apparent shape and extent is positively influenced by sound localisation accuracy; this is normal since subjects relied on the position of the decorrelated sound sources to perceive the apparent source extents or shapes.

Experiments 5 (sections 4.8) showed that the use of broad sound sources improved the naturalness of 3D audio scenes and experiments 6 (sections 4.8) showed that dynamic decorrelation could also improve realism, however it can lead to listening fatigue.

In conclusion, this research showed that it is possible to render and perceive the extent and the apparent shapes of sound sources with noise signals, however, time varying and complex signals such as music are still problematic. In section 6.2 are

highlighted a solution to this problem and areas where the research described in this chapter could be furthered. A practical implementation of sound source extent rendering in a real-time 3D audio rendering system is described in section 5.4.4.

Chapter 5

Implementation of an object oriented 3D audio scene renderer

5.1 Introduction

This chapter presents the implementation aspect of this thesis by describing the new 3D audio rendering system known as Configurable Hemispheric Environment for Spatialised Sound (CHESS). This system was researched and implemented by the author for the purposes of experimentation during this thesis work.

CHESS is a novel real-time system for rendering and composing 3D audio scenes based on the XML scene description scheme that was described in chapter 3. Based on a client-server architecture and novel configurable 16-speaker array, CHESS is aimed at rendering 3D audio scenes to a small audience (3-4 people). The architecture and techniques used in CHESS can be used as a model to implement subsequent 3D audio rendering systems. CHESS also implements the sound source extent rendering algorithms described in chapter 4.

This chapter first considers the global system overview of CHESS and shows how CHESS follows the server-client architecture. This explains why the client-server approach was selected over other approaches. The client is a Java3D program that

consists of the system user interface and the scene manager and updater. The server consists of a Digital Signal Processing layer that implements the necessary DSP tasks to render the 3D audio scenes.

The signal processing layer, implemented in Max/Msp [Max], is then described. CHESS follows a hybrid perceptual and physical approach to render 3D audio scenes; and a justification of this approach is given. The individual DSP tasks are then detailed and a discussion of certain 3D audio techniques over others given. The Java3D scene manager client is then described. It is shown how this module is responsible for parsing 3D audio scene XML description and updating and managing the scene.

An evaluation of the system is then given. Finally, practical applications and major projects where CHESS has been used are reviewed.

5.2 CHESS system overview

The overall system overview of CHESS is shown in Fig. 5.1 where the functions and technologies used for the different system parts are defined. CHESS is composed of three main modules: the scene manager, the DSP layer and the speaker array.

The scene manager, acting as a client, instantiates scene objects from the XML 3D audio scene description (chapter 3) at the system DSP layer. During play-back of the scene, the scene manager then updates the state of the scene at the DSP layer by sending commands on the network (section 5.5). Based on Java and Java3D, the scene manager also collects real-time user commands through a user interface and provides a 3D graphical representation of the 3D audio scene being rendered.

Implemented on a separate computer and acting as the server, the DSP layer of CHESS performs all necessary signal processing tasks to render the 3D audio scene to the user(s). The DSP layer is detailed in section 5.3. The communication between the scene manager and the DSP layer is established using the UDP [RK] and Open Sound Control (OSC) [osc] protocols over an ethernet network.

The use of the client-server approach allows the global computational load to be shared by two computers. This approach also allows the DSP layer to be controlled remotely from any computer connected on the Internet. The CHESS system could thus be used for 3D audio teleconferencing and remote collaboration applications. The client-server approach has been used in other 3D audio rendering systems [CETT02, Sch02], in which the client-server approach is used to distribute the 3D audio signal processing tasks between multiple computers. When using a distributed DSP layer, extra care must be taken to insure that the output channels of the systems are perfectly synchronised. In the CHESS system, all processing is done on the same machine, and so this insures that all output audio channels are synchronised.

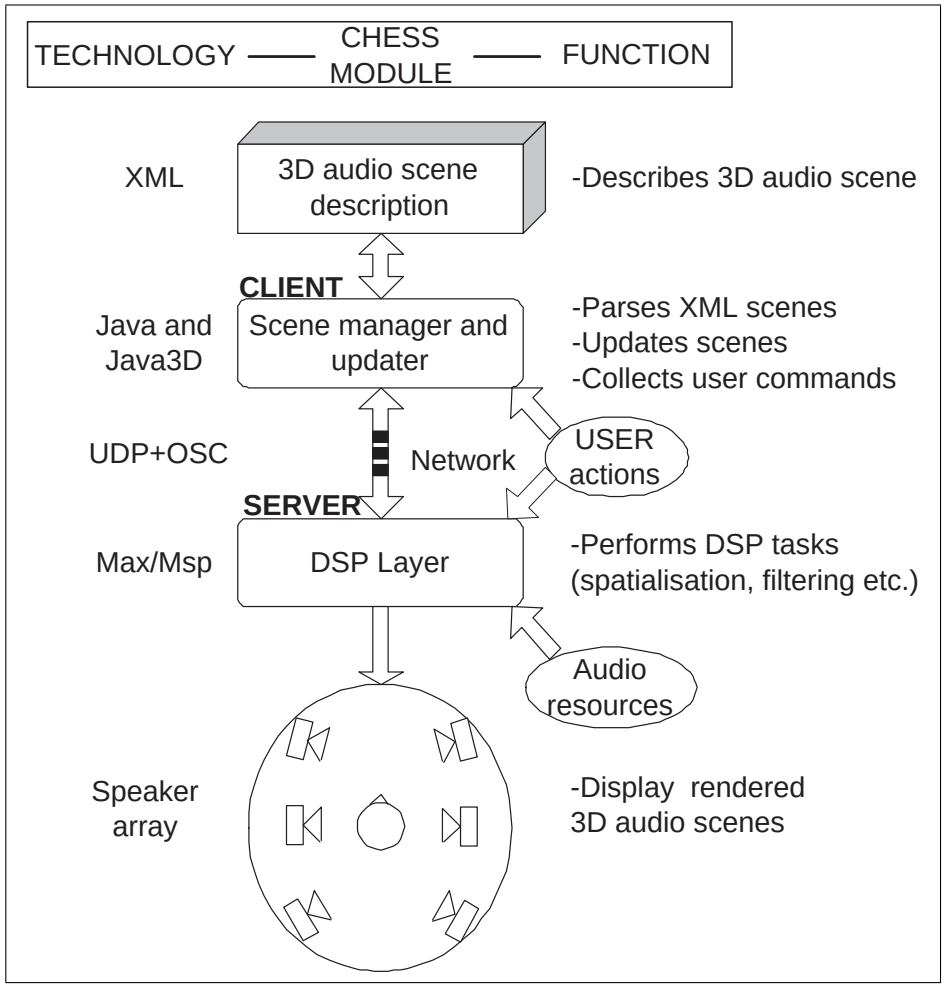


Figure 5.1: Overview of the client-server structure of the CHESS system and the functions and technologies of the different system parts

5.2.1 Speaker vs headphone 3D audio rendering

To render 3D audio scenes, speaker based spatialisation was selected since binaural headphone techniques [Beg92a, Car96], despite some efforts [MT02], are aimed at individual use. CHESS is aimed at being experienced by few people at a time and headphone spatialisation is unfit for this task, since users are separated in their own 3D audio environments and head-tracking devices are required for each system user.

When speaker spatialisation is used, the user's own Head Related Transfer Functions (HRTF) are used to localise the position of sound sources, this contrasts with

headphone rendering where measured or generic HRTFs must be used, resulting in localisation errors [Beg94]. Furthermore, it was estimated in [JLP99] that binaural techniques which require a convolution operation and a head-tracking device [WDO97] are ten times more computationally expensive than speaker based spatialisation.

Transaural spatialisation techniques [Bau93, KJM03, KTH99] are binaural techniques applied on speakers. These techniques, however, still require a high processing cost and do not allow for multiple user scenarios since the 'sweet spot'[1] is limited to a small area. A divergence of just a few centimeters from the sweet spot by the user destroys the 3D audio impression [MRK00]. For these reasons, transaural spatialisation techniques were also discarded.

Speaker-based spatialisation was thus selected so that CHESS provides an immersive environment which can be experienced by several simultaneous users. To provide a large sweet-spot area where listener can perceive the correct rendered 3D audio soundfield, a high order Ambisonics spatialisation technique was selected. Daniel [Dan03a] showed that the size of the sweet-spot increased with Ambisonics order, thus the highest possible order was selected for CHESS. The spatialisation technique used in CHESS is further detailed in section 5.4.1.

5.2.2 CHESS speaker array

The speaker array used in CHESS is depicted in Fig. 5.2. The 16-speaker array is attached on a novel configurable scaffold which permits the speakers to slide up and down, and be placed and oriented them anywhere around the centre of the sphere where a small audience (3-4 people) can be accommodated. The speaker array was designed by the author and Didier Balez[2] who carried out the construction work. This configurable scaffold structure allows to quickly change and test different speaker configurations. Sixteen high quality monitoring speakers (Genelec 1029A) were used

[1] That is, where the user must be located to perceived the 3D audio scene correctly

[2] Didier Balez is head of the sculpting workshop at the Faculty of Creative Arts, University of Wollongong

for the speaker array. Using a spherical arrangement guaranteed that each speaker was equidistantly placed from the center of the array, which is a requirement for Ambisonics spatialisation used by CHESS (section 5.4.1). The speaker array has a diameter of approximately 3m.

A 3D audio system with a configurable speaker array is also proposed in [KKFW99]; in this system, speakers are hung from the ceiling.

Figure 5.2: The configurable speaker array of the CHESS system

5.2.3 Hardware

The Java3D scene manager program was implemented on a Pentium III PC. The DSP layer is implemented on a Macintosh G4 867MHz computer connected to a Digi001 soundcard and an additional ADAT digital to analog converter so that the DSP layer is capable of outputting 16 audio channels at 44.1 or 48 kHz sampling frequency with a 16-bit precision. The use of a low-latency operating system for the DSP layer (Mac

OS9) allowed sound card latency to be kept under 15ms.

5.3 Digital signal processing layer

The Digital signal processing layer of CHESS performs all the necessary tasks to render 3D audio scenes. It is controlled via the network by the scene manager (section 5.5) or can be operated locally by the user via a graphical user interface (GUI) depicted in Fig. 5.3.

In the proceeding sections, the selection of the Max/Msp platform for implementing the DSP layer is first explained, follows the description of the global DSP chain of the CHESS system. The individual signal processing tasks are then explained. In the description of these various tasks, the selection of certain techniques is explained.

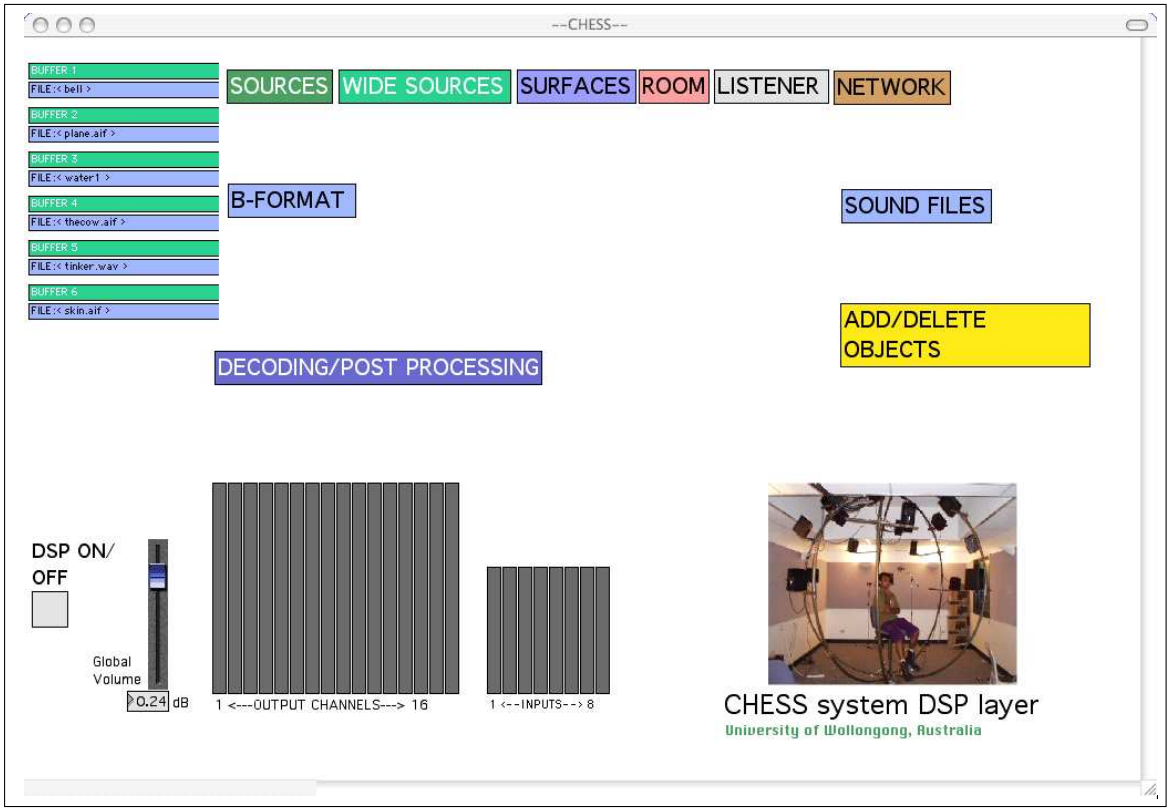


Figure 5.3: Graphical user interface of the DSP layer

5.3.1 Selection of the rendering platform

Several 3D audio rendering systems [DRS+03, HDM03] use consumer class 3D audio APIs such as OpenAL [Ope], Creative EAX [EAX] or Sensaura [sen]. Relying on a particular 3D audio API has several drawbacks. Firstly, 3D audio APIs are usually targeted towards video games and consumer electronic applications and, being mostly based on binaural and transaural spatialisation, do not offer accurate multi-speaker spatialisation capabilities.

Secondly, some commercial 3D audio APIs (e.g. EAX) rely on special hardware implementations which are likely to become obsolete when a new soundcard or operating system becomes available. Therefore, developing a 3D audio rendering system that is heavily based on a particular API can be a risk. Lastly, the fine DSP implementation details of the APIs are usually not available for intellectual property reasons. The author believes that a 3D audio rendering system implementation that is free of hardware or API dependance provides greater control in the fine details of the 3D audio rendering.

Other 3D audio rendering systems based on hardware implementations (such as the Lake Huron audio workstation [hur]) have been quickly surpassed by the ever increasing power of general purpose CPUs. A 3D audio rendering system developed in software may be moved to a faster machine, while an hardware implementation requires implementation on different DSP chips.

The Max/Msp platform

To implement the DSP layer, the Max/Msp [Zic02, Max] graphical programming language was used. This platform has been used to develop other 3D audio rendering systems [KKFW99, JW95, Lun00, Sea03]. Thanks to its graphical programming environment a lot of prototyping and design time may be saved. In addition, it is highly efficient for performing audio DSP and some objects for performing 3D audio rendering are readily available [JW95, Pul97]. Max/Msp can also handle large

numbers of input and output channels[3] using low latency ASIO drivers; this proved to be useful when implementing the DSP layer of the CHESS system.

Although less efficient than a pure C++ implementation used in some systems [Nae02, HST96, MW02] the Max/Msp platform efficiency proved to be sufficient for the required task. Besides, when higher efficiency is required, Max/Msp has provisions for using compiled objects written in C or C++. This feature was used to implement decorrelation filters[4], to calculate reflections[5] and sound source occlusions[6]. An overview of the DSP layer is now given.

5.3.2 3D audio signal processing overview

To implement the 3D audio rendering layer, a hybrid physical and perceptual approach was used. The physical approach is used to spatialise sound sources, calculate reflections and to simulate sound propagation in air. The perceptual approach is used to compute room reverberation, sound source occlusion effects and sound source extent rendering. This hybrid approach is also used in other 3D audio rendering systems such as DIVA [HSHT96] and "Le Spatialisateur' [JW95]. However these systems always describe sound sources as point sources, this differs in CHESS where the extent of the sound sources can be controlled.

Another novel part of CHESS is the use of an 'Ambisonic bus' (Fig. 5.4) which collects all the spatialised sound sources. This architecture allows saving the composed 3D audio scenes in an Ambisonics encoded form which can be later decoded on a different speaker configuration. The 'Ambisonic bus' can also be used to import 3D audio scenes that were previously composed or were recorded by a soundfield microphone (section 2.3.3). This allows the creation of hybrid 3D audio scenes that are the combination of recorded and synthetic 3D audio scenes. Ambisonic spatialisation is detailed in section 5.4.1. The use of Ambisonics in CHESS contrasts with other

[3] up to 512!
[4] section 5.4.4
[5] section 5.4.7
[6] section 5.4.6

system such as DIVA [HST96] or SLAB [MW02] which either use binaural or speaker array dependant spatialisation algorithms (e.g. VBAP see section 5.4.1) and are thus less versatile.

An overview of the CHESS global signal processing architectures is shown in (Fig. 5.4). The DSP tasks are grouped in three categories: calculation of the direct sound source signal reaching the listener, calculation of specular reflections and simulation of late reverberation.

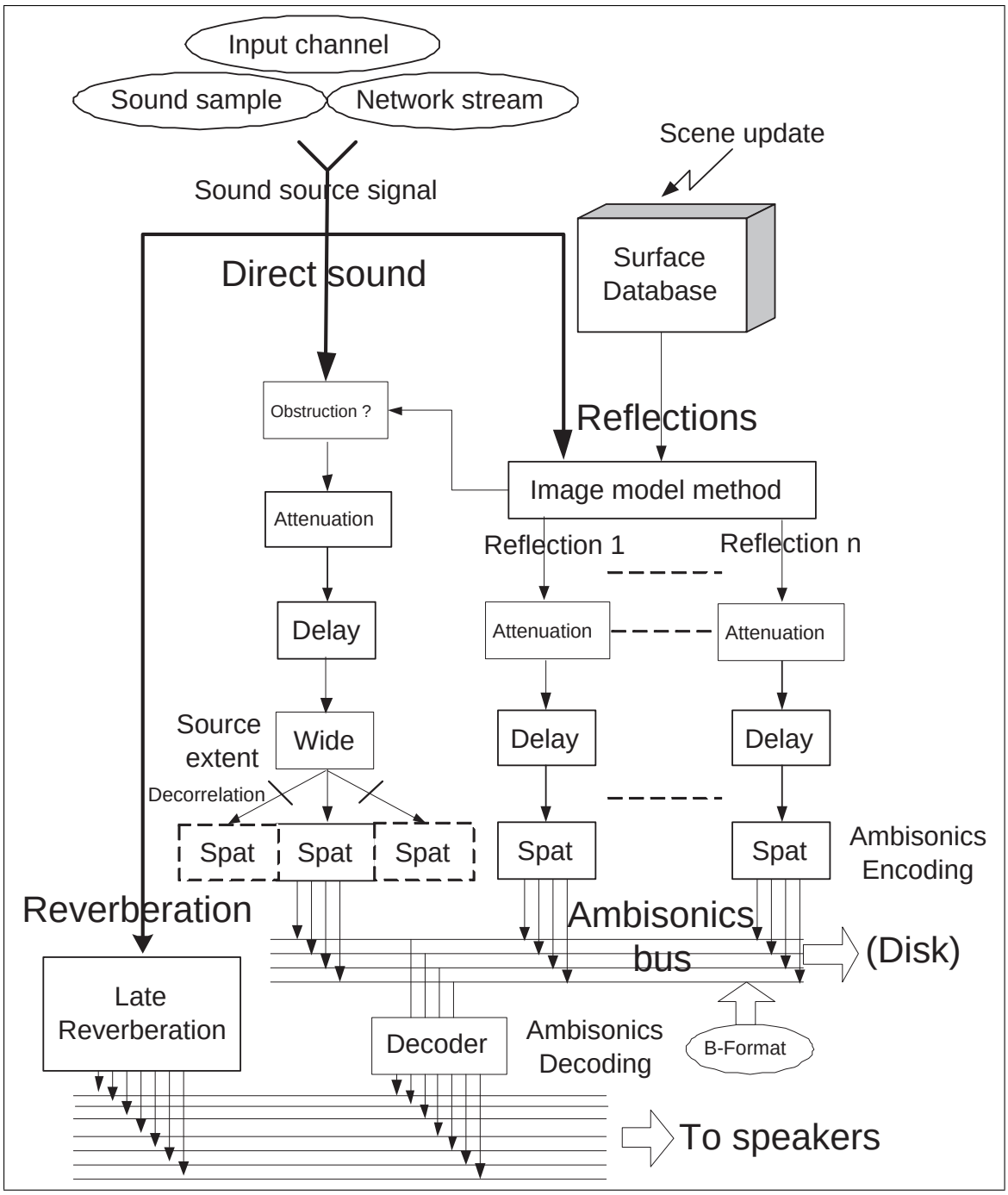


Figure 5.4: Overview of the signal processing chain in CHESS for the calculation of direct sound, reflections and reverberation for one sound source

The calculation of the direct sound source signal, calculation of specular reflections and simulation of late reverberation are now described.

Direct sound source signal

The direct sound source signal processing chain simulates the direct propagation of a sound source towards the listener. This is based on a source-medium-receiver model [HST96] which consists in modelling aspects of the virtual sound source (e.g. its position, size etc.), the propagation properties of the medium (e.g. delay, attenuation, obstructions) and aspects of the listener (e.g. his/her position and orientation). Another sound source propagation model is the 'Room within the room model' developed by Moore[Moo83], however this model also defines spatialisation[7] and so Ambisonics spatialisation and the 'Ambisonics bus' of the CHESS system could not have been employed. This model was thus discarded since its use would have reduced the versatility of the CHESS system.

The signal processing chain used to calculate the direct sound source signal is shown in Fig. 5.5. The direct sound signal is first attenuated in the case where the path between the source and the listener is obstructed by the presence of an obstacle; the detection of such sound source occlusion is explained in section 5.4.6. The source signal is then attenuated and delayed to simulate sound propagation in the virtual medium and the distance of the sound source (section 5.4.3 and 5.4.5). The extent or size of the sound source is then rendered (section 5.4.4). Lastly, the final sound source signal is spatialised (section 5.4.1). The directivity of sound sources has not been implemented for computational limitation reasons (see section 5.6), however there should be no difficulty in implementing this in later versions of CHESS; one approach would be to use a directivity filter [SHLV97] in the path of the direct sound source signal.

[7] Which is equivalent to amplitude panning

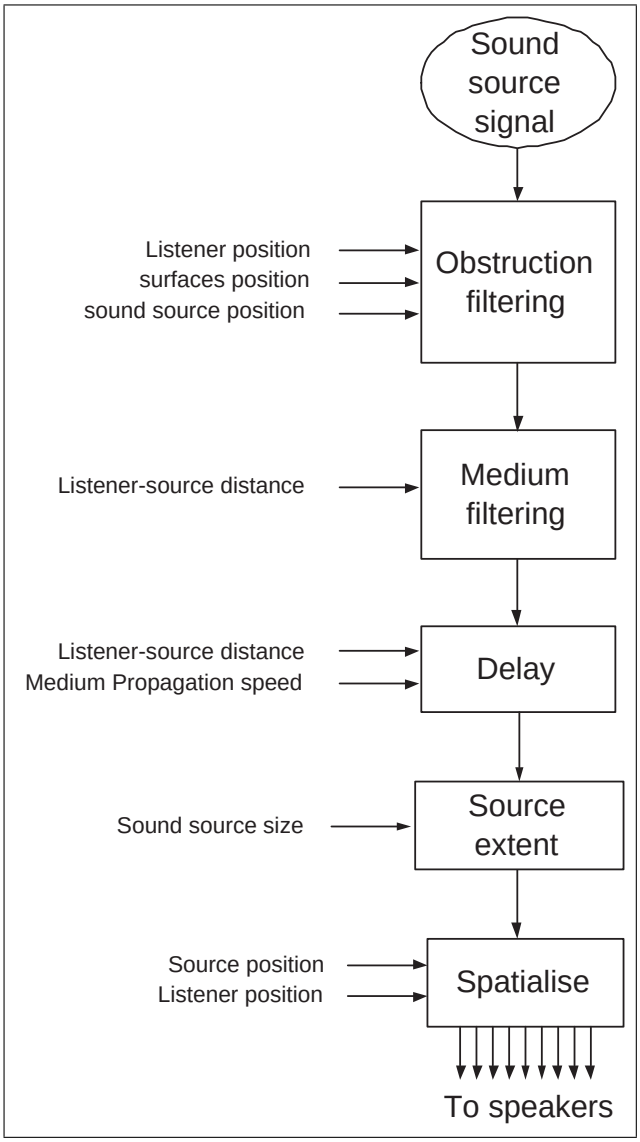


Figure 5.5: Signal processing chain for the direct signal path

Specular reflections

Specular reflections occur when virtual surfaces present in the scene reflect sound source signals towards the listener. If the surfaces define the walls of a room, this can be used to compute the early reverberation of the room. In CHESS, reflections are calculated using an image model algorithm (section 5.4.7). This algorithm consists of simulating sound reflections by spatialising phantom sound sources. Each phantom sound source is then attenuated, delayed and spatialised according to its distance and position in the 3D audio scenes (Fig. 5.4 right). For computational limitation reasons, only first order[8] reflections are calculated in order to limit the total number of sound sources to be processed (i.e. direct and phantom sound sources).

Reverberation

Late reverberation is simulated using a Feedback Delay Network (FDN) [RS97] driven by perceptual parameters. This approach is detailed and justified in section 5.4.8. The separation of the early reflection and late reverberation computation is a typical approach in real-time acoustical simulation [Sch70, SHLV97]. Indeed, the calculation of sound reflections using only a physical approach requires a very high processing load since the number of reflections grows exponentially and reaches several million within only a few seconds of the original sound.

Another approach to simulate room reverberation is to use convolution of the source signal with a measured impulse response [hur], however convolution is a computationally expensive process and impulse responses must be measured for different source and listener positions in the room [HSHT96].

[8] That is, reflections of reflections are not computed

5.4 Description of 3D audio processing tasks used in CHESS

The different Digital Processing tasks used to render 3D audio scenes in CHESS are now detailed. These are:

- Spatialisation of sound sources (section 5.4.1 and 5.4.2)
- Rendering of sound source distance (section 5.4.3)
- Rendering of sound source extent (section 5.4.4)
- Simulation of propagation delays and Doppler effect (section 5.4.5)
- Detection and simulation of sound source occlusion (section 5.4.6)
- Calculation of specular reflections and early reverberation (section 5.4.7)
- Simulation of late reverberation (section 5.4.8)

5.4.1 Spatialisation

Spatialisation is the action of processing a monaural signal so that it is perceived by a listener as emanating from a virtual sound source located at a certain position. To perform spatialisation on speakers, three approaches were available: amplitude panning techniques, Wave Field Synthesis Techniques and Ambisonics techniques.

Wave Field Synthesis (WFS) Techniques [Boo95, Ber88, VB99, Baa03] are able to create virtual sound sources with great precision however a dense array of speakers is required wherever virtual sound sources are to be spatialised. Thus in general, the WFS technique is used only for 2D horizontal spatialisation. For instance, the WFS rendering system at Delft University (The Netherlands) [Uni] uses more than two hundred speakers to achieve only surrounding 2D spatialisation. Two-dimensional

spatialisation is unable to represent the elevation of virtual sources, and thus full 3D audio immersive fields cannot be reproduced.

Since CHESS aims at reproducing full enveloping 3D audio scenes and there was a limit in the number of speakers that could be purchased, WFS spatialisation was discarded.

Amplitude panning techniques [Wes98, Pul01, Ger92e, DFMM98] such as the Vector Based Amplitude Panning VBAP [Pul97] consist in using stereo panning laws [Bau61, Blu33] between pairs or triplets of speakers. This requires the a priori knowledge of the coordinates of the speakers. In contrast, the Ambisonics technique [Dan00, Ger98b, Ger98a, Mal95] is able to encode 3D audio content that is independent of the speaker configuration (section 2.3.3).

It was also highlighted in [Pul99] that VBAP suffers from fluctuations in sound source extent during source displacement due to a variable number of speakers used at a time to perform source panning. For these reasons, Ambisonics spatialisation was finally selected. Ambisonics has however the drawback that a regular speaker array should be used, unlike VBAP which can be used on any arbitrary speaker configurations. This however was not an issue since a regular geodesic dome speaker array was used; the selection of the speaker array geometry is explained in section 5.4.1. Another advantage of the Ambisonics is that 3D content produced with CHESS may be saved in Ambisonics form (speaker independent) and can be later exported and played on various speaker configurations.

The Ambisonics technique in its first order form is described first and then in its higher order form. The implementation of the Higher Order Ambisonics (HOA) technique in CHESS is then detailed.

First order Ambisonics

In its first order form, Ambisonics encodes a monaural source signal $s(t)$ at azimuth θ and elevation δ into four signals representing the spatialised sound source. These four signals are known as the B-format signal [Ger85, Mal95].

$$\left| \begin{array}{l} W(t) = s(t) \\ X(t) = s(t) \cdot cos(\theta) \cdot cos(\delta) \\ Y(t) = s(t) \cdot sin(\theta) \cdot cos(\delta) \\ Z(t) = s(t) \cdot sin(\delta) \end{array} \right| \tag{5.1}$$

Fig. 5.6 shows the azimuth θ and elevation δ of a sound source, this spherical coordinate convention is also used in CHESS.

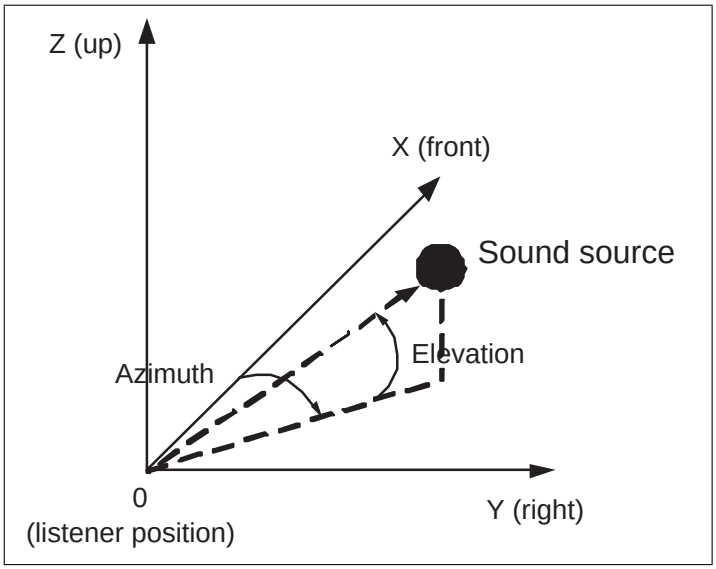


Figure 5.6: Spherical coordinate system used in the CHESS system

Since Ambisonics spatialisation is a linear process [Dan00], a 3D audio scene may be composed by adding several B-format signals together. The scene is then decoded to the target speaker configuration via a single decoding matrix; this was explained in section 2.3.3 and depicted in Fig. 2.6. Decoding equations for various speaker configurations (cube, hexagon, octagon etc.) can be found in [FM]. B-format may also be decoded to a 5.1 standard speaker arrangement [Ger92a].

First order Ambisonics is a simple technique which is computationally efficient since few multiplications and trigonometrical operations are needed for each signal

sample. The size of the sweet spot, however, only allowed room for one user at the centre of the speaker array. Alternate Ambisonics decoding techniques can be used to increase the size of the sweet spot area [Mal92]. Even better results can be obtained with Higher Order Ambisonics (HOA) as it has been mathematically demonstrated [Dan03a, NE98] that HOA increases the sweet spot area and improves the sharpness and localisation of spatialised sound sources. This lead to the final choice of using HOA spatialisation in CHESS. The HOA encoding and decoding operations are now described.

Higher Order Ambisonics encoding

To implement Higher Order Ambisonics spatialisation used in CHESS, a mathematical formulation of HOA developed by Daniel [Dan00] was used. Ambisonics relies on the decomposition into spherical harmonics functions of the sound pressure field generated by a sound source of azimuth θ and elevation δ received at the centre point[9].

In order to encode a monaural source signal to a desired position with azimuth θ and elevation δ, a $y(\theta, \delta)$ vector containing the values of the $Y_n(\theta, \delta)$ spherical harmonics functions needs first to be computed [Dan00]:

$$y(\theta, \delta) = [Y_1(\theta, \delta), Y_2(\theta, \delta), Y_3(\theta, \delta), Y_4(\theta, \delta), \ldots, Y_{(m+1)^2}(\theta, \delta)] \tag{5.2}$$

The length of this encoding vector equals $(m + 1)^2$ where m is the Ambisonics order [Dan00]. To obtain the Ambisonics encoded signals $B(t)$, the monaural signal $s(t)$ to be spatialised at position (θ,δ) is multiplied by the vector $y(\theta, \delta)$ containing the encoding coefficients:

$$B(t) = s(t) \cdot y(\theta, \delta) \tag{5.3}$$

[9]After decoding, this point corresponds to the central point of the speaker array

Thus $(m+1)^2$ Ambisonics encoded signals are obtained. The Ambisonics encoding operation is illustrated in Fig. 5.7. First order Ambisonics requires four channels (i.e. B-format), second Ambisonics requires 9 channels, 4th order Ambisonics requires 25 channels etc. The higher the order, the higher the number of Ambisonics channels are required to describe the sound field at one point. A perfect sound field description would require an infinite number of spherical harmonics functions [Dan00]. Practically, the Ambisonics order needs to be limited so as to limit the number of Ambisonics signals. The maximum usable Ambisonics order is also limited by the number of speakers of the array, this is explained in section 5.4.1.

The $Y_n(\theta, \delta)$ spherical harmonics encoding functions up to order 4 are given in table 5.1. Functions up to order 3 were obtained from [Dan00] and functions for order 4 were calculated by the author.

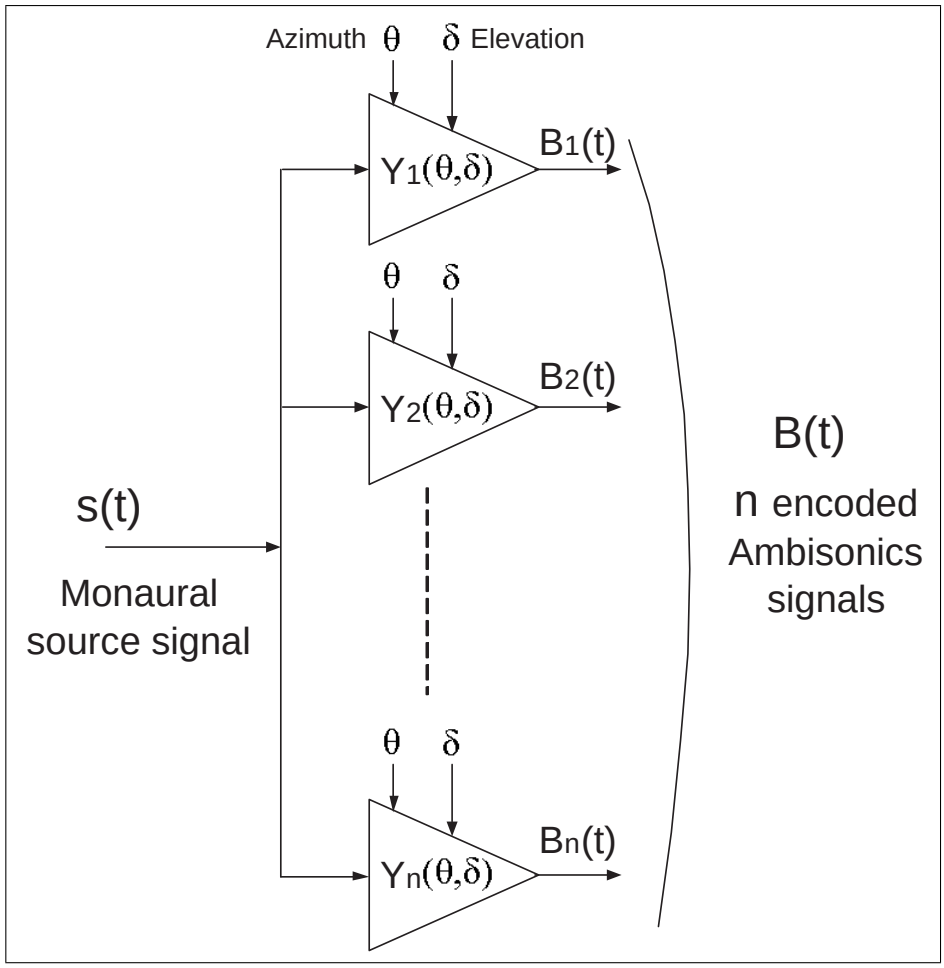


Figure 5.7: Illustration of the Higher Order Ambisonics encoding operation

Order	Channel ID	n	$Y_n(\theta, \delta)$
0	W	1	1
	X	2	$\cos(\theta)\cos(\delta)$
1	Y	3	$\sin(\theta)\cos(\delta)$
	Z	4	$\sin(\delta)$
	R	5	$\frac{\sqrt{3}}{2}\cos(2\theta)\cos^2(\delta)$
	S	6	$\frac{\sqrt{3}}{2}\sin(2\theta)\cos^2(\delta)$
2	T	7	$\frac{\sqrt{3}}{2}\cos(\theta)\sin(2\delta)$
	U	8	$\frac{\sqrt{3}}{2}\sin(\theta)\sin(2\delta)$
	V	9	$(3\sin^2\delta - 1)/2$
	K	10	$\sqrt{\frac{5}{8}}\cos(3\theta)\cos^3\delta$
	L	11	$\sqrt{\frac{5}{8}}\sin(3\theta)\cos^3\delta$
	M	12	$\sqrt{\frac{15}{2}}\cos(2\theta)\sin\delta\cos^2\delta$
3	N	13	$\sqrt{\frac{15}{2}}\sin(2\theta)\sin\delta\cos^2\delta$
	O	14	$\sqrt{\frac{3}{8}}\cos\theta\cos\delta(5\sin^2\delta - 1)$
	P	15	$\sqrt{\frac{3}{8}}\sin\theta\cos\delta(5\sin^2\delta - 1)$
	Q	16	$\sin\delta(5\sin^2\delta - 3)/2$
	A	17	$\sqrt{\frac{70}{128}}\cos(4\theta)cos^4\delta$
	B	18	$\sqrt{\frac{70}{128}}\sin(4\theta)cos^4\delta$
	C	19	$\sqrt{\frac{105}{24}}\sin\delta\cos^3\delta\cos(3\theta)$
	D	20	$\sqrt{\frac{105}{24}}\sin\delta\cos^3\delta\sin(3\theta)$
4	E	21	$\sqrt{\frac{5}{16}}(7\sin^2\delta - 1)\cos^2\delta\cos(2\theta)$
	F	22	$\sqrt{\frac{5}{16}}(7\sin^2\delta - 1)\cos^2\delta\sin(2\theta)$
	G	23	$\sqrt{\frac{5}{8}}\sin\delta(7\sin\delta - 3)\cos^2\theta$
	H	24	$\sqrt{\frac{5}{8}}\sin\delta(7\sin\delta - 3)\cos\theta\sin\theta$
	I	25	$\frac{1}{8}(35\sin^4\delta - 30\sin^2\theta + 3)$

Table 5.1: Spherical harmonics encoding equations up to Ambisonics order 4, (Encoding equations up to order 3 obtained from [Dan00])

Ambisonics is a linear process, and several encoded sound source signals may be summed channel by channel to produce a set of Ambisonics signals containing the whole sound scene. Then a single Ambisonics decoder is required to render the scene on speakers. This is illustrated in Fig. 5.8.

$$B_{scene}(t) = \sum_{n=0}^{totalsources} B_n(t) \tag{5.4}$$

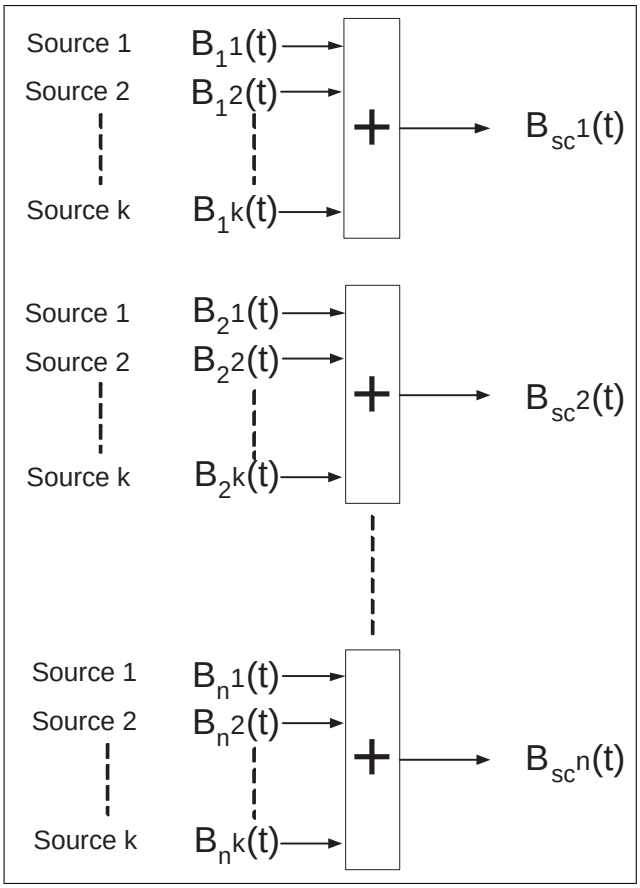


Figure 5.8: Illustration of the forming of an audio scene by adding n Ambisonics signals from k encoded sound sources

Higher Order Ambisonics decoding

The Ambisonics decoding operation can be summarised by finding a matrix D that transforms the encoded signals $B(t)$ into the sought speaker signals $S(t)$. This is illustrated in Fig. 5.9.

$$S(t) = D.B(t) \tag{5.5}$$

The contribution of each speaker to the reconstructed sound field B' can be written as:

$$B'(t) = C.S(t) \tag{5.6}$$

Where C is called the re-encoding matrix [Dan00] and is defined as:

$$C = [c_1, c_2, \ldots, c_i, \ldots, c_N] \tag{5.7}$$

Where i is the speaker index and N the total number of speakers and c_i the vector containing the values of the $Y_i(\theta_i, \delta_i)$ spherical harmonics functions defined in equation 5.2

$$y(\theta_i, \delta_i) = [Y_1(\theta_i, \delta_i), Y_2(\theta_i, \delta_i), Y_3(\theta_i, \delta_i), Y_4(\theta_i, \delta_i), \ldots, Y_{(m+1)^2}(\theta_i, \delta_i)] \tag{5.8}$$

Where θ_i and δ_i are the azimuth and elevation of the i^{th} speaker.

From equation 5.5 and 5.6 and posing $B = B'$, it can be seen that the decoding matrix D is the pseudo-inverse of the re-encoding matrix C [Dan00]:

$$D = pinv(C) = C^T \cdot (C.C^T)^{-1} \tag{5.9}$$

In order to perform the pseudo-inverse operation, there should be enough speakers [DRP98]; the maximum usable Ambisonics order for the 16-speaker array of CHESS is 4. This is the reason why Ambisonics 4th order is used in CHESS. Alternatively, the 16 speakers of CHESS can be used to render lower order Ambisonics. An advantage of the Ambisonics technique is that whenever a speaker array is unable to render a particular Ambisonics order[10] (due to a low number of speakers or a irregular configuration), it is possible to discard the higher order Ambisonics component and to decode only the lower ones; making Ambisonics truly scalable.

The pseudo-inverse operation is non-trivial for irregular speaker arrays and thus high order Ambisonics is usually used with regular speaker arrays. It can be checked whether a speaker configuration is regular in the Ambisonics way by verifying that the re-encoding matrix C satisfies the following equation:

$$\frac{1}{N} C \cdot C^t = I_k \tag{5.10}$$

Where I_k is the diagonal unitary matrix of rank k and $k = (m+1)^2$ (i.e. the number of encoded Ambisonics signals).

[10] For instance, Ambisonics orders greater than four for CHESS

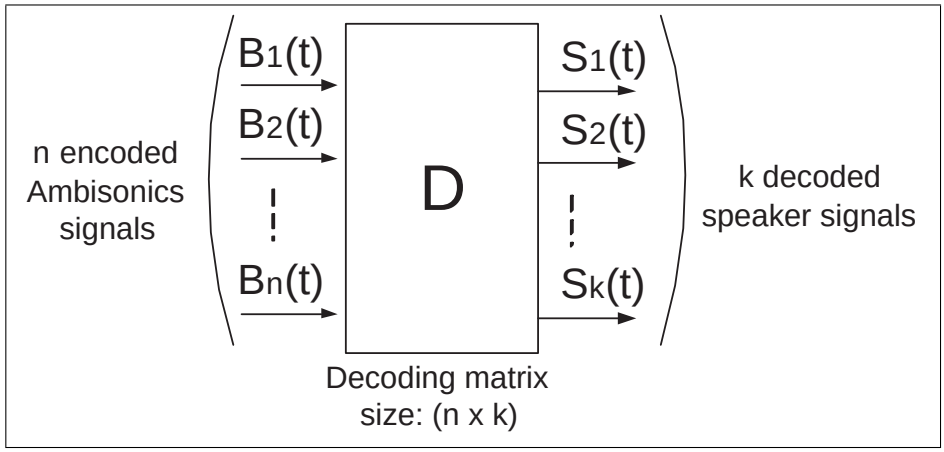


Figure 5.9: Diagram of the Ambisonics decoding process via a decoding matrix D

Selection of the speaker array used in CHESS

In order to perform Ambisonics spatialisation, the speaker array must be regular (i.e. satisfy equation 5.10) in order for the decoding matrix D to be found from the re-encoding matrix C (equation 5.9). Since sixteen speakers were available to build the CHESS speaker array, the only regular configuration with sixteen speakers consists of the vertices of a half dodecahedron polyhedra; this is also called a geodesic dome configuration. The dodecahedron polyhedra is depicted in 5.10; the CHESS dome speaker array consist of the upper hemisphere of this polyhedra.In a later stage CHESS could be extended to 32 speakers by using all the vertices of the dodecahedron polyhedra.

The first obvious consequence of using only the upper hemisphere, is that virtual sources cannot be rendered below subjects. In real life however, it is rare to find sound source emitting at such locations (except for floor reflections), thus this limitation was not considered to be an issue for CHESS. Another consequence of truncating the speaker array to the upper hemisphere is that boundary effects appear when virtual sources are spatialised close to the horizon and in between speakers. Boundary effects produce diffuse and unstable spatialised sound sources because of the missing speakers

of the lower hemisphere; boundary effects were also observed by Daniel [Dan00]. In the case of CHESS, however, boundary effects could be easily avoided by spatialising virtual sources with a slight elevation (e.g. 5 or 10 degrees), this insured that missing speakers of the lower hemisphere had a minimal effect.

In terms of decoding the Ambisonics signals for 16 speakers (upper hemisphere) instead of 32 (full sphere), the decoding matrix D defined in equation 5.9 is simply truncated to a $n \times \frac{k}{2}$ sized matrix where n is the number of Ambisonics signals and $\frac{k}{2}$ the number of speakers (here $\frac{k}{2} = 16$).

The azimuth and elevation of the speakers were calculated with the program WinDome [win]. The speaker coordinates were then used to calculate the decoding matrix D at Ambisonics order 4. The speaker array diagram is depicted in Fig. 5.11 and the coordinates of the speakers are given in table 5.2.

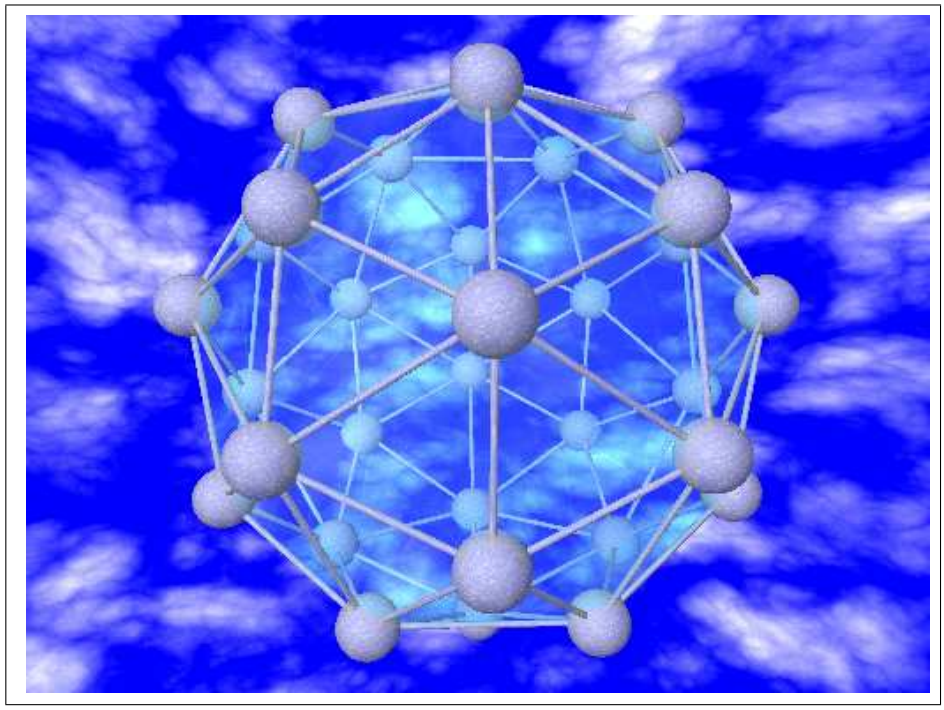


Figure 5.10: Diagram of the icosahedron polyhedra used to place speakers (upper hemisphere used only)

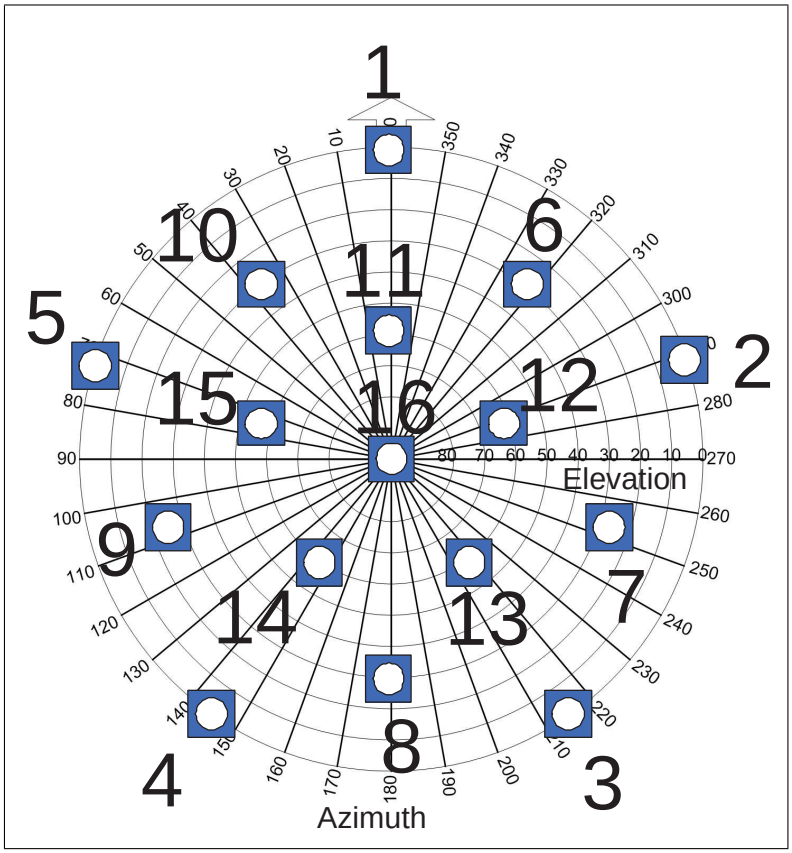


Figure 5.11: Numbering and placement of speakers in CHESS (top-down view)

Speaker #	Azimuth	Elevation
1	0	10.8
2	72	10.8
3	144	10.8
4	216	10.8
5	288	10.8
6	36	26.6
7	108	26.6
8	180	26.6
9	252	26.6
10	324	26.6
11	0	52.6
12	72	52.6
13	144	52.6
14	216	52.6
15	288	52.6
16	0	90

Table 5.2: Coordinates of the CHESS speakers

5.4.2 Implementation of 4th order Ambisonics spatialisation in CHESS

From the HOA theory described in section 5.4.1, new 4th order Ambisonics encoding and decoding Max/Msp objects were created; these objects were then used in the implementation of the DSP layer of CHESS.

The HOA encoding Max/Msp object was implemented by calculating the encoding gains (described in table 5.1) at the wanted *Azimuth* and *Elevation* of the sound source. The source signal is then multiplied by these gains to obtain a set of encoded Ambisonics signals. The new HOA encoding Max/Msp object is depicted in Fig. 5.12.

The HOA decoding Max/Msp object was implemented using the native *matrix~* object of Max/Msp filled with the coefficients of the decoding matrix D. The new HOA decoding Max/Msp object is depicted in Fig. 5.13.

The encoding and decoding HOA objects were also ported to VST plugins [vst] using the Pluggo [plu] environment. This allowed HOA spatialisation to be performed directly within audio sequencing programs such as Protools, Logic Audio etc. The user interface of the 4th order Ambisonics encoding VST plugin is shown in Fig. 5.14.

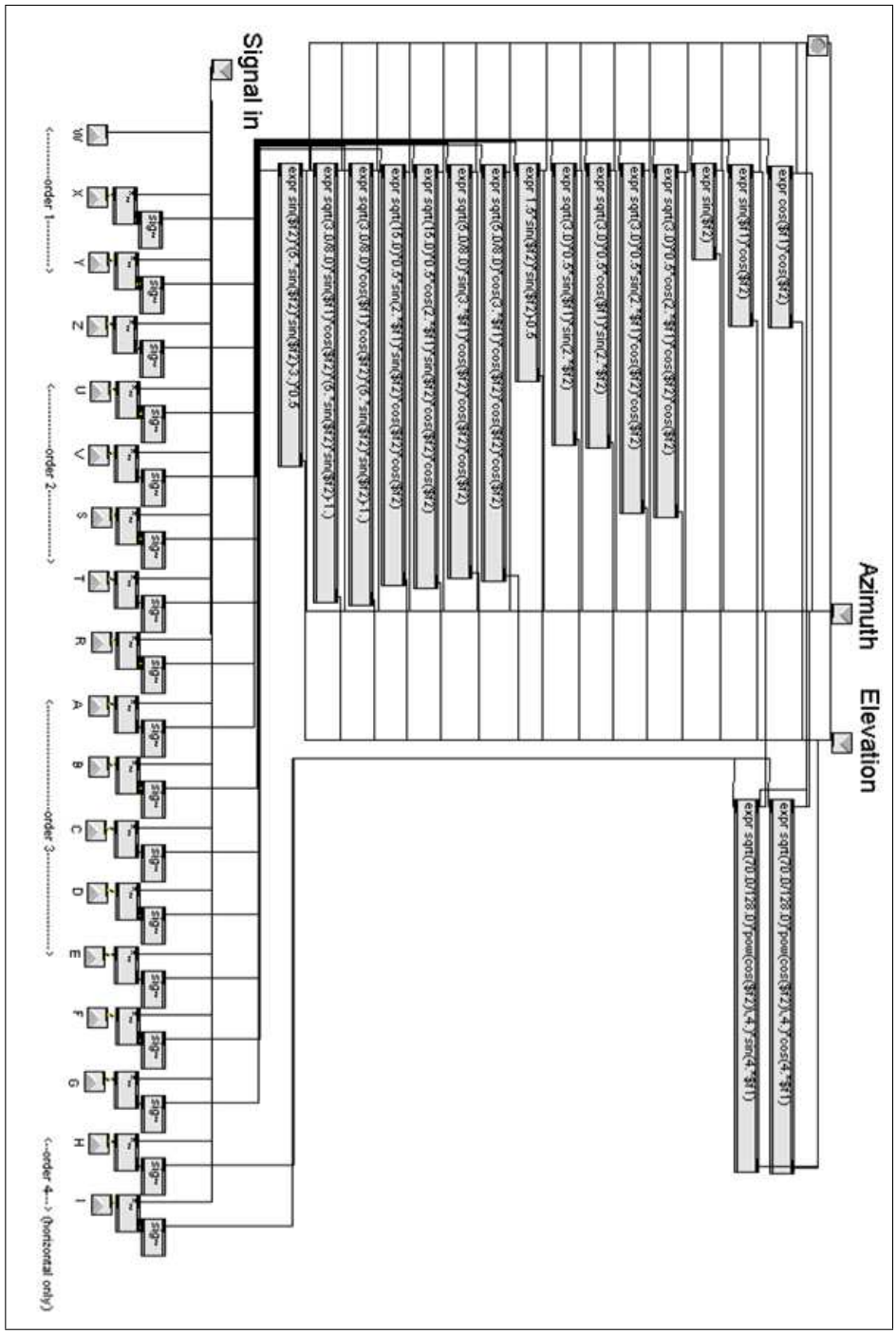


Figure 5.12: Patch of the new Max/Msp object for performing 4th order Ambisonics encoding in CHESS

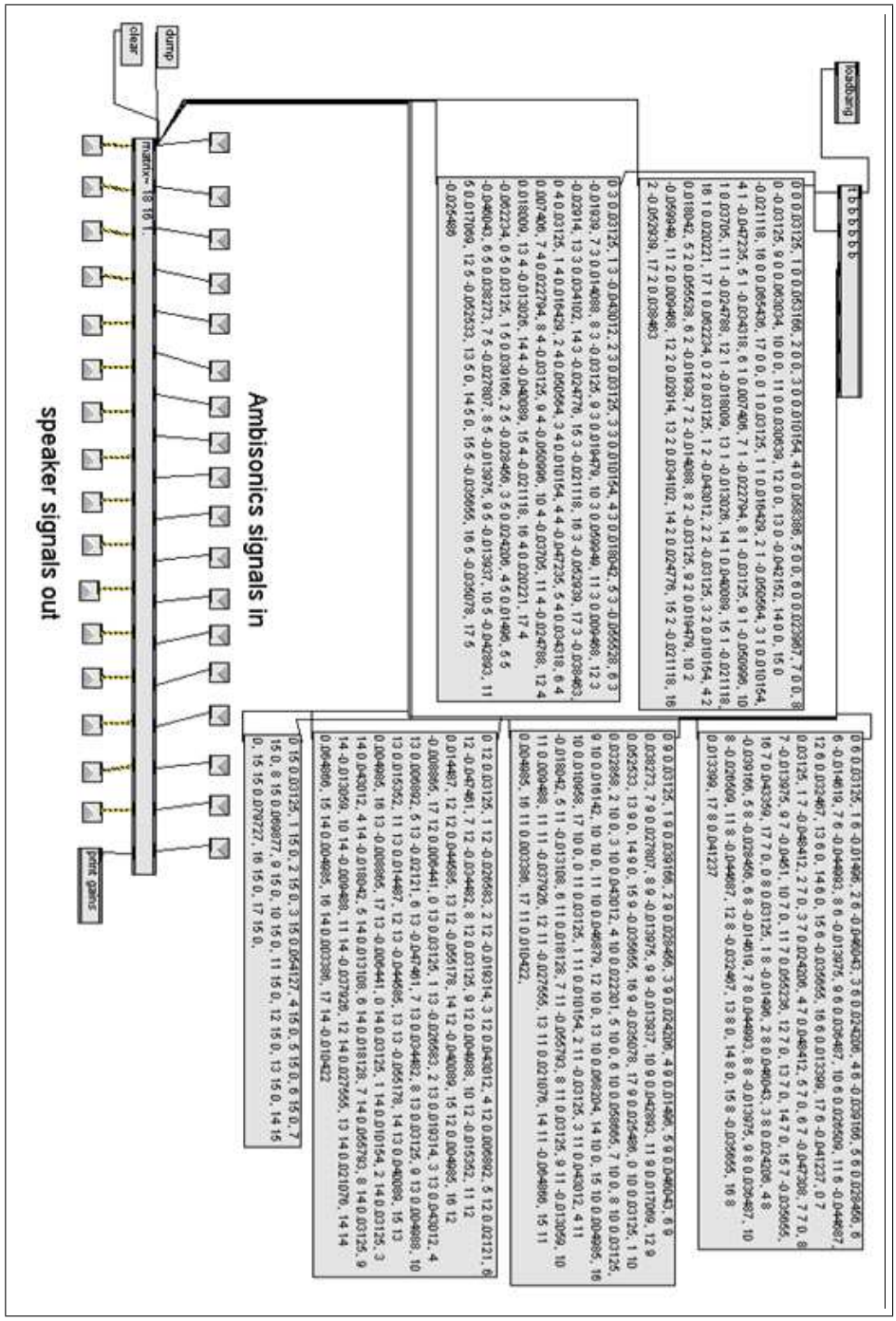


Figure 5.13: Patch of the new Max/Msp patch for performing 4th order Ambisonics decoding in CHESS

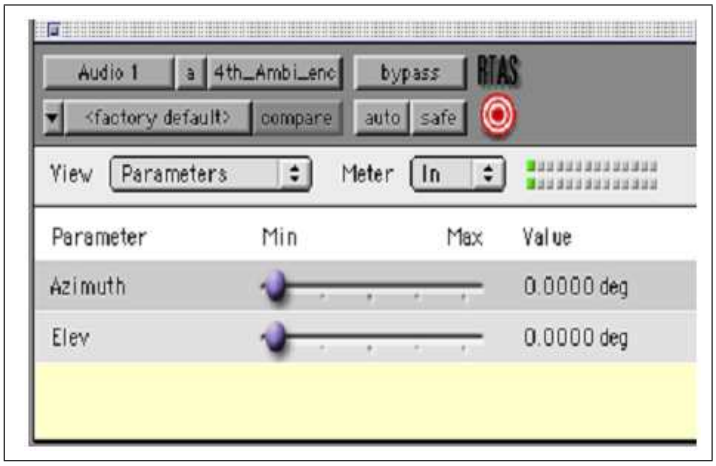


Figure 5.14: Interface of the 4th Order Ambisonics spatialisation plugin in Protools

5.4.3 Sound source distance rendering

The perceptual cues used to appreciate sound source distance by listeners were reviewed in section 2.5.2. These cues are: the diminution of source intensity with distance, the low-pass filtering caused by air attenuation, the reverberant to direct sound ratio (R/D), and the curvature of the wavefront.

Physical approach to source distance rendering

Some spatialisation techniques such as Wave Field Synthesis (WFS) [Boo95, Ber88] and the recently developed Ambisonics distance coding [Dan03a, Dan03b, SH01] are able to control sound source distance directly by synthesising the sound field as it would normally be produced by a natural sound source. For instance, the wavefront of a sound source will be rendered as being planar for a distant sound source and rendered as being curved for a close sound source. These techniques thus reproduce sound source distance in a physical way. Physical control of sound source distance is however reliable only at short distances from the listener where the binaural system uses the curvature of the wavefront to detect binaural differences [Zah02][11]. In the

[11]Near-field effects also produce a bass-boost (section 2.5.2)

far-field (i.e. approximately after two meters), wavefront curvature cues are not used [Zah02] and thus, the physical approach to source distance rendering is not usable.

As far as Ambisonics distance coding is concerned, this technique can be used to create virtual sound sources *inside* the speaker array; this is impossible with traditional Ambisonics. However to the knowledge of the author, no subjective evaluation of this technique was carried out. Ambisonics with distance coding could be implemented in future developments of CHESS.

A last technique which can be used to produce virtual sound sources within the speaker array is the Audio Spotlight [spo] technique which is based on an ultrasound beam that becomes audible after travelling a few meters in the air; thus creating the stable illusion of a virtual sound source at a location where there is nothing (at the point in space where parts of the ultrasound beam spectrum are shifted in the audible range due to non-linear distortion caused by air). This technique could be used in conjunction with Ambisonics or amplitude panning techniques to place sound sources within the speaker array.

Perceptual approach to source distance rendering

Since the spatialisation technique that is used (HOA) in CHESS does not allow control of source distance, a perceptual approach to distance control was used; this is an often used approach in virtual auditory display and sound studio techniques [Cho71, Ger92d, OFR02, FRO02, Zah02]. This approach allows placing sound sources from the surface of the speaker array (minimal distance) to large distances. The perceptual approach is unable to produce near-field effects such as wave front curvature and thus the minimal distance d_{min} where a sound source can be placed corresponds to the distance between the speakers and the listener. This is illustrated in Fig. 5.15.

The perceptual rendering of sound source distance in CHESS is performed by three means: attenuation of the source signal with distance, emulation of air filtering effects and reverberation.

The intensity attenuation gain G_d due to the propagation of the sound source

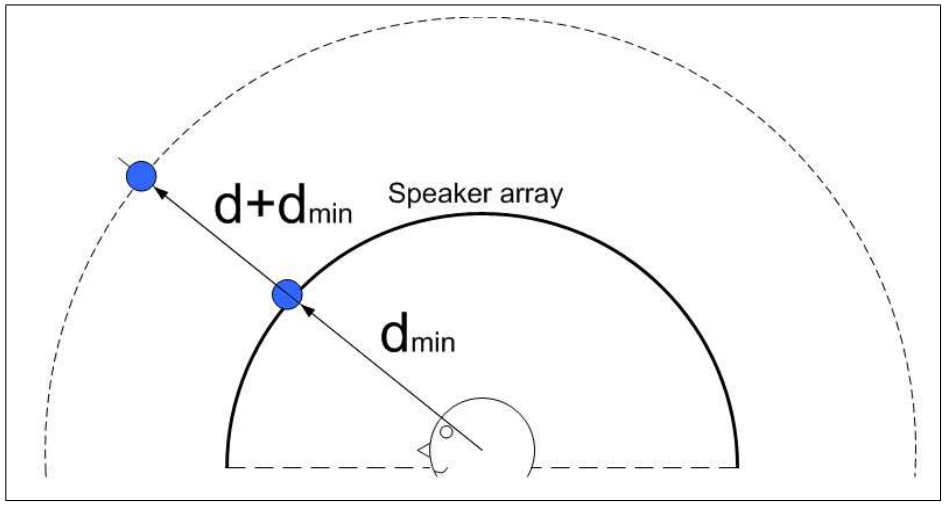


Figure 5.15: Illustration of distance control and the minimum source distance in CHESS

signal in the virtual medium is inversely proportional to the source distance:

$$G_d = \frac{d_{min}}{d_{source} + d_{min}} \tag{5.11}$$

Where d_{min} is the distance between the listener and the speaker array and d_{source} the position of the sound source from the speaker array (see Fig. 5.15). Therefore if the sound source is placed at the minimum distance, no attenuation occurs.

The source signal $s(t)$ is then attenuated by G_d:

$$s'(t) = s(t) \cdot G_d \tag{5.12}$$

Where $s'(t)$ is the new source signal to be spatialised or passed down the DSP chain (Fig. 5.5).

If artificial room reverberation is used, this in turn, increases the reverberant to direct sound (R/D) ratio automatically. The (R/D) ratio is the most reliable psychoacoustic cue in determining source distance, thus sound source distance perception ability is notoriously better in reverberant than in anechoic environments

[Nie93, Wag90].

To complete the rendering of source distance, the source signal is filtered by an air attenuation filter. The description of air attenuation coefficients in function of air temperature and humidity can be found in [Har66, BSZ95, BB72] and a simplified model can be found in [HSK97]. Practically, this consists of low-pass filtering the source signal with a filter, the high frequency cut-off of which is increased with source distance.

To implement air attenuation in CHESS the Max/Msp *air~* object from the Spat [JW95] object library is used; this relies on a simplified model similar to [HSK97].

5.4.4 Sound source extent rendering

The perception and rendering of sound source extent was detailed in chapter 4. In CHESS, sound source extent is rendered using the decorrelated point source technique which was detailed in 2.12. This technique creates decorrelated signal replicas of the original monaural source signal; these are then fed to virtual sound sources spatialised at different locations. This creates a broad sound source which extent is defined by the position of the point sound sources.

In CHESS, the point sources were spatialised using 4th order Ambisonics (section 5.4.2), however the number of point sources used to create a broad sound source was limited to three due to computational limitations. This however, gave an homogeneous broad source image if the extent was not greater than 90 degrees. If source extent was increased beyond 90 degrees, this resulted in a loss of binaural fusion[12] and the individual point sources could be localised separately. This is due to the density of the point sources which, when too low, permits the individual localisation of the point sources. This effect was also discovered and explained in a psychoacoustic experiment described in section 4.3. Therefore, to create broader sound sources, more decorrelated point sources must be used so that the density of point sources remains higher than one source per 30 degrees (see section 4.3).

[12] binaural fusion was described in section 2.8.5

The positioning of the point sources was performed as follows: let θ_1 be the Azimuth of the centre of the sound source and W the *Width* parameter; the azimuths of the two other decorrelated sound sources was then given by $(\theta_1 + W/2)$ and $(\theta_1 - W/2)$. This is depicted in Fig. 5.16.

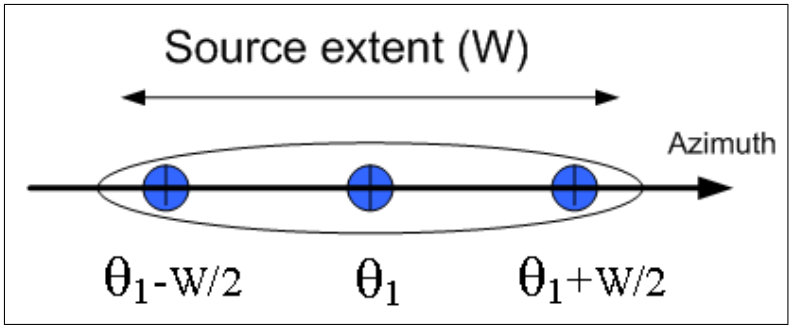


Figure 5.16: Illustration of simple horizontal source extent rendering in CHESS

To create decorrelated signals in real-time, new Max/Msp externals[13] were developed that produced 3, 6 and 8 decorrelated signals. Decorrelation was performed via all-pass IIR filters of order 50, implemented in Direct Form II [Mit03] to improve coding efficiency. The order of the decorrelation filters was chosen as a compromise between decorrelation strength and computational efficiency; order 50 for the IIR filters provided good decorrelation while not imposing too much computational load on the DSP layer.

The stable, all-pass IIR filter coefficients were calculated in Matlab[14] and were hardwired in the Max/Msp object source code. The new decorrelation object is shown in Fig. 5.17.

5.4.5 Propagation delays and Doppler effect

To simulate the delay caused by sound propagation in the virtual medium, the sound source signal to be spatialised is digitally delayed according to:

[13]That is, Max/Msp objects that are programmed in C language and then compiled
[14]From Matlab code that was developed and which can be found in Annex 7.2

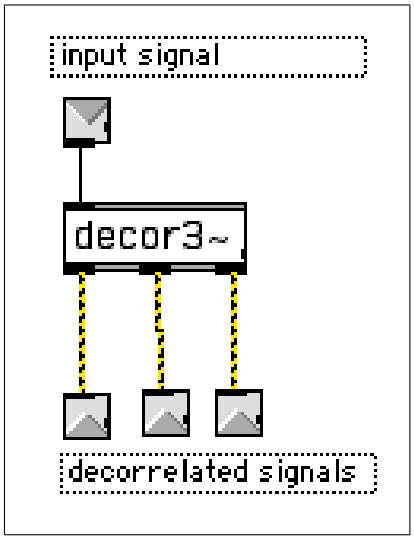


Figure 5.17: New decorrelation object for Max/Msp

$$t_{delay} = \frac{d_{source}}{c} \tag{5.13}$$

Where d_{source} is the source distance (in meters) and c the celerity of the medium (in m/s). A constant propagation speed of 304 m/s was used.

Implementation of the source delay is done in a variable digital delay line controlled by the delay t_{delay}. The advantage of using a variable delay line is that Doppler effects are automatically produced during changes in distance between the listener and the sound source; this is a well known property of variable delay lines [Moo83, SSA02, Nae02].

A variable digital delay line is implemented by a circular buffer[15] and write and read pointers. The diagram of a variable delay line is shown in Fig. 5.18. By changing the rate at which the read or write pointer progresses depending on the speed of the listener and the sound source respectively, the pitch and play back speed of the read signal are modified, akin to Doppler effect. In early Doppler effect implementations, variable delay lines were created on a magnetic tape loop with moving read and write heads [SSA02].

[15] which length depends on maximum permissible delay and available memory

The pitch shifting caused by changes in distance can be defined as follows: the frequency f_l received by the listener of a sound source emitting a frequency f_s is equal to [SSA02]:

$$f_l = (\frac{c}{c + v_s}) \cdot f_s \tag{5.14}$$

Where c is the speed of sound, v_s the relative speed of the sound source to the listener (v_s is negative for approaching sound sources and positive for receding sources).

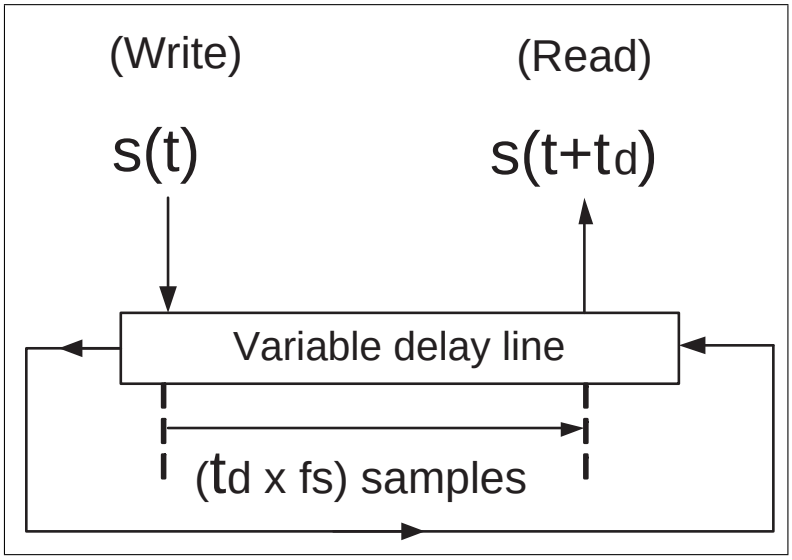


Figure 5.18: Diagram of a variable delay line to implement delay and Doppler effects

To implement delay and Doppler effects in CHESS, the native Max/Msp variable delay line *vdelay~* object was used. This object was then controlled in real time by the calculated distance between the sound source and the listener. This gave satisfactory impressions of Doppler effect when sound sources were moved away and towards the listener at great speed. This in turn improved the realism of the rendered 3D audio scenes, since the Doppler effect is an important perceptual cue in the perception of sound source movements [Cho71, Moo83].

5.4.6 Sound source occlusion

To simulate the sound muffling caused by the presence of an obstructing surface between the listener and the sound source (Fig. 5.19), the algorithm described in Fig. 5.20 was developed. This technique is heavily inspired by the image model algorithm [AB79] described in section 5.4.7.

The algorithm first detects if the vector $\overrightarrow{SL}$ joining the sound source with the listener intersects with the plane of the surface. If it does, the intersection point A with the surface plane is calculated. Then, a test is performed to determine whether the intersection point is within the vertices (V_1, V_2, V_3, V_4) of the surface.

If after determining that a surface is occluding the visibility of a sound source from the listener point of view, a simple low-pass filtering is applied on the sound source signal to simulate the muffling caused by the sound source obstruction. The low-pass filter was implemented using a biquad IIR architecture with a cut-off frequency of 500 Hz as it proved to be realistic sounding. The selected approach for emulating sound occlusion is a simplification of the technique used in [HST96, HSHT96] where different absorbing materials can be defined.

No diffraction effects were taken into account as described in [FBAA03] since these act as sound sources present at the edges of the obstructing surface; creating a shift in sound source position. It was decided that muffling the sound source signal to emulate sound source occlusion was more perceptually relevant than emulating diffraction effects. The obstruction test must be performed between any sound source and surface present in the scene. Therefore, to limit computational complexity, the definition of surfaces was limited to four vertices. For the same reason, only a few surfaces and sound sources may be used at one time in the rendered scene.

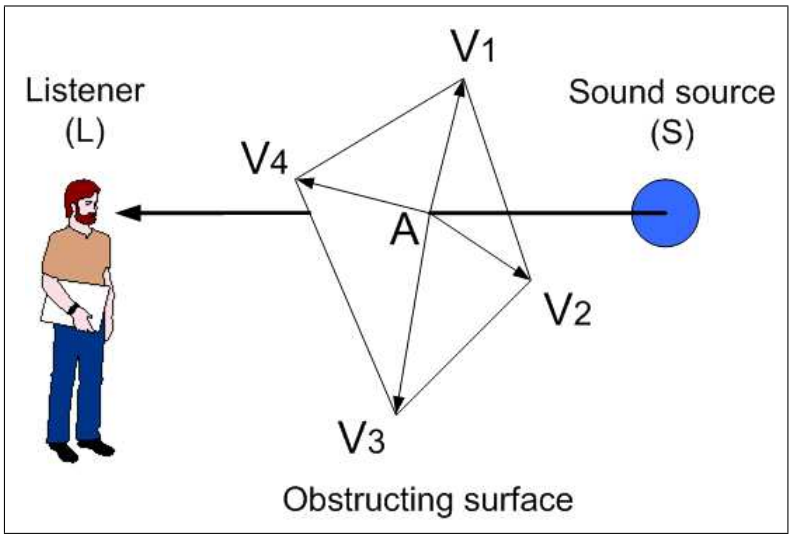


Figure 5.19: Detection of sound source occlusion by a surface object

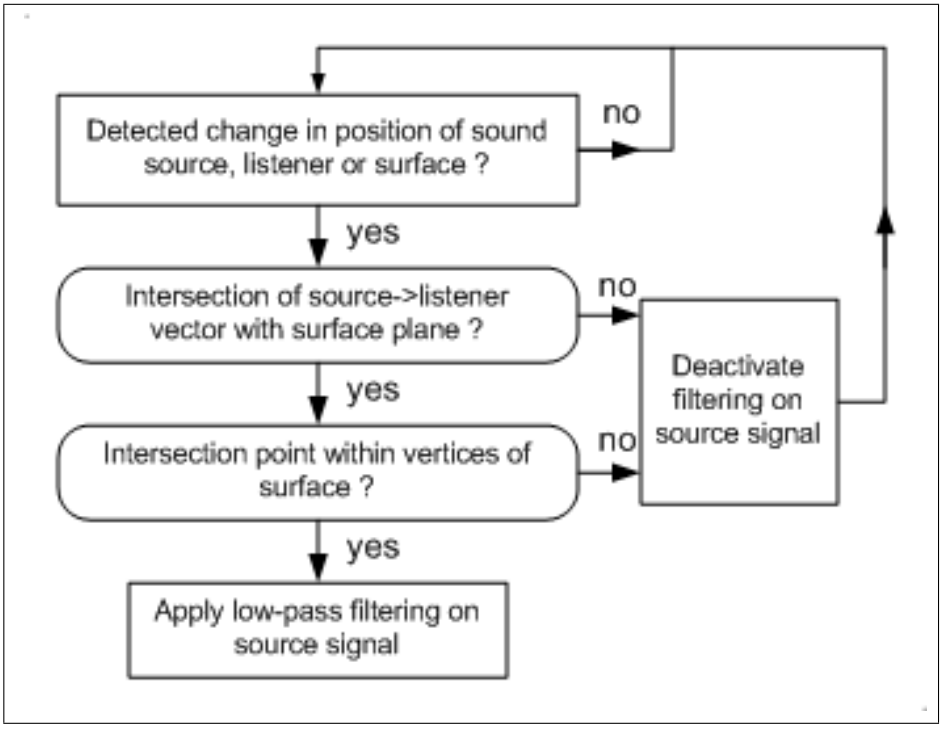


Figure 5.20: Algorithm for sound source occlusion detection in CHESS

5.4.7 Early reflections calculation

In order to calculate the specular reflections of the source signals on the reflective surfaces present in the 3D audio scene, an image model algorithm is used [LL88, AB79, Bor84]. The image model algorithm is based on the observation that a reflected sound source is equivalent to a phantom sound source located *behind* the reflective surface. The principle of the image model algorithm is shown in Fig. 5.21.

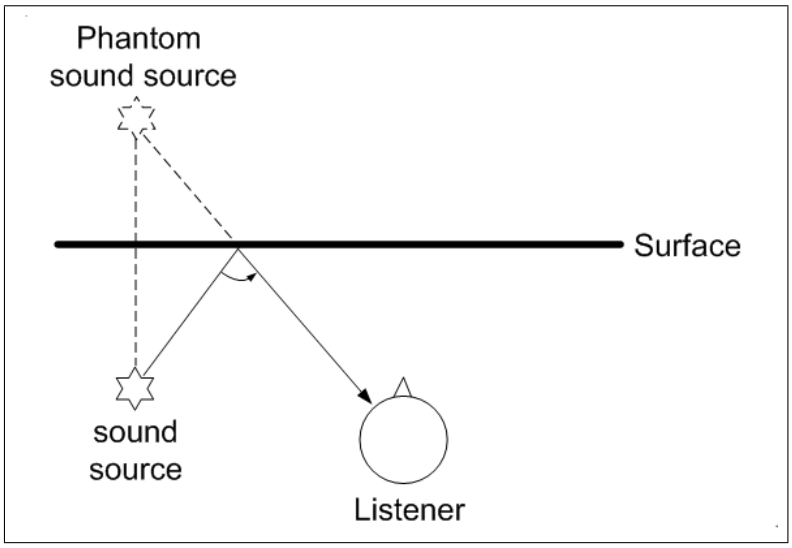


Figure 5.21: Illustration of the image model algorithm principle

This algorithm was selected as it allowed to emulate reflections by directly spatialising sound sources on the Ambisonics bus (section 5.3.2). The ray-tracing [Vor89] or beam-tracing [Dru97, FPST03] algorithms use a different model where many rays are emitted by sound sources and reflections computed; rendering this approach impractical with the Ambisonics bus architecture used by CHESS. Some beam-tracing approaches [FTCa04, FMC99, FCE98] are efficient enough to be suitable for real-time applications.

The image model algorithm is depicted in Fig. 5.22 and performs as follows: when movement of the sound source, surfaces or listener is detected, the algorithm calculates the new positions of the phantom sources caused by each of the surface. The calculated phantom source is first tested for validity, that is, if the phantom source

is placed on the same side of the surface as the listener, it is considered invalid. If the phantom source passes the validity test, it is tested for visibility from the current listener position; this is done by checking whether the path between the listener and the phantom sound source intersects with the surface.

When the phantom sound source both passes validity and visibility tests, a last test is performed to check whether another surface is occluding the sound source; this test is also carried out for the direct sound and was described in section 5.4.6. Finally the phantom source is spatialised in the 3D audio scene; the phantom source signal is also attenuated by a fixed attenuation coefficient and delayed according to its distance. A fixed arbitrary reflection attenuation coefficient of 0.5 was used to limit computational load. With more processing power, materials with frequency dependant attenuation coefficients could be emulated; reflection coefficients for various material can be found in [Sen99]. Since the image algorithm had to be carried out between any source and surface pair of the scene, the number of these had to be limited to a few. In most cases, six surfaces were used to form the walls of a room. Only first order reflections were considered, that is, reflections of reflections are not computed so as to simplify computation.

It was found that the image model gives satisfactory results for reflections of distant surfaces perceived as distinct echoes; however, since only first order reflections are calculated, this technique is not able to produce late reverberation. Late reverberation is generated by another module which is reviewed in section 5.4.8. The image model algorithm was implemented in a Max/Msp external object to be used in CHESS.

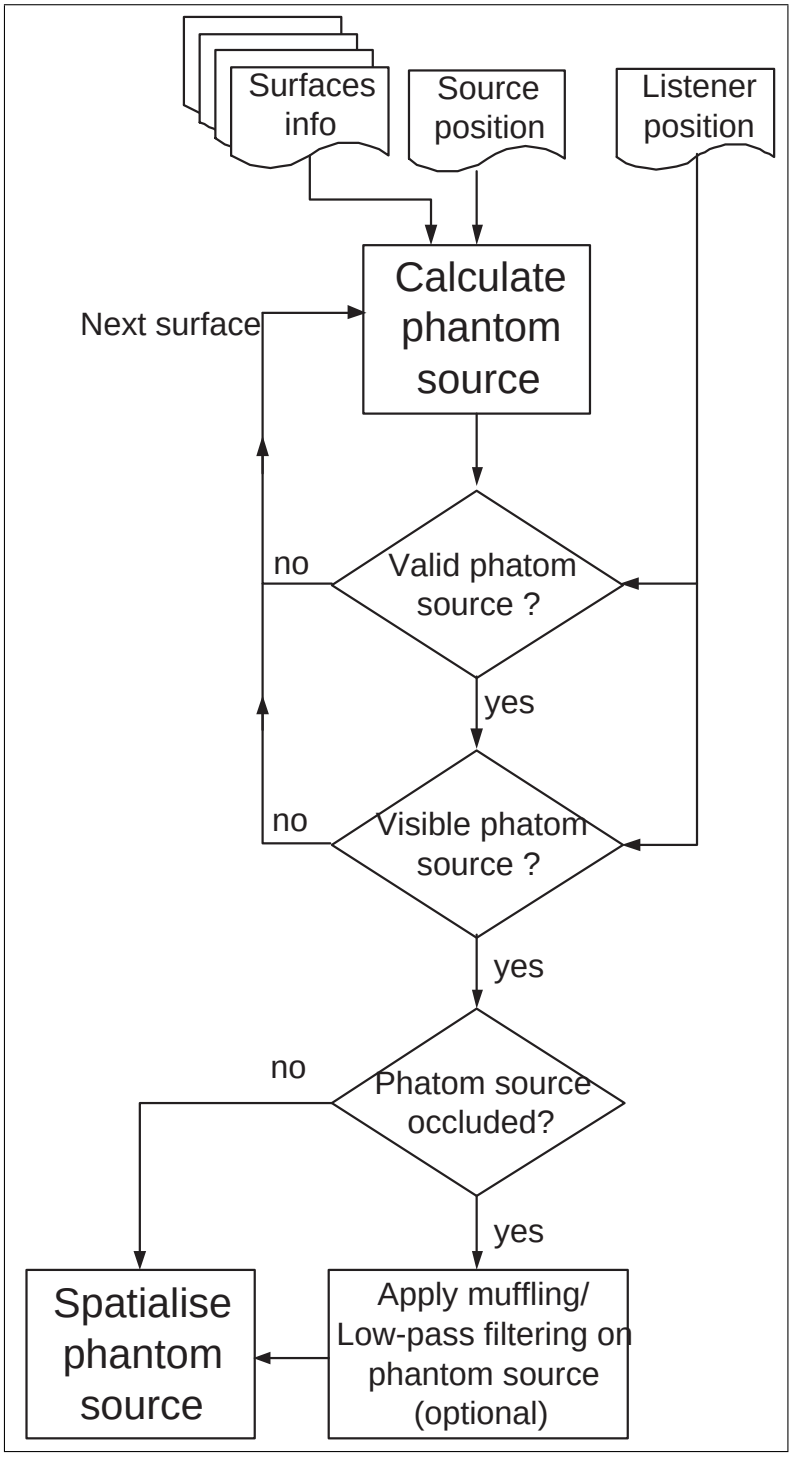


Figure 5.22: Diagram of the first order image model algorithm used in CHESS

5.4.8 Late reverberation

The simulation of late reverberation is useful as it improves the realism and quality of rendered 3D audio scenes by providing a sensation of spaciousness and envelopment [Pel01, Beg92b] (section 2.9) and placing the listener is a certain acoustic context [Pel00]. The use of reverberation also greatly improves the rendering of sound source distance (section 5.4.3). Digital room reverberation has been an important research topic and early models [Sch70, Moo79] used comb and all-pass filters to simulate multiple echoes. Later models use Feedback Delay Networks (FDN) [SP82, RS97, GT02, Roc95].

In CHESS, room reverberation is performed by FDN architecture which is controlled by perceptual parameters [Jot97]. This approach was selected as it provides good sounding reverberation which could be controlled by a few parameters and allowed creating virtual acoustic environments ranging from a small room to large reverberant halls. The perceptual approach to room reverberation is also used in MPEG-4 Advanced AudioBIFS (section 2.4.2) and in IRCAM's Spatialisateur [JW95].

To implement room reverberation on a multi speaker system such as CHESS, a multi-channel reverberator must be used since the use of multiple stereo reverberators would produce correlated reverberation signals between speakers, resulting in a higher Inter-aural Cross Correlation coefficient[16], in turn defeating the impression of spaciousness (chapter 4) and degrading the quality of reverberation. In CHESS, the signals of the sound sources present in the rendered 3D audio scene are summed and fed to a 16-channel reverberation module which then outputs decorrelated reverberated signals directly to each speaker (Fig. 5.23).

The perceptual parameters and the graphical interface used to control room reverberation in CHESS are detailed in Fig. 5.23. The *Spat* Max/Msp objects [JW95] were used to implement the room reverberation module.

[16]Defined in section 2.8.2

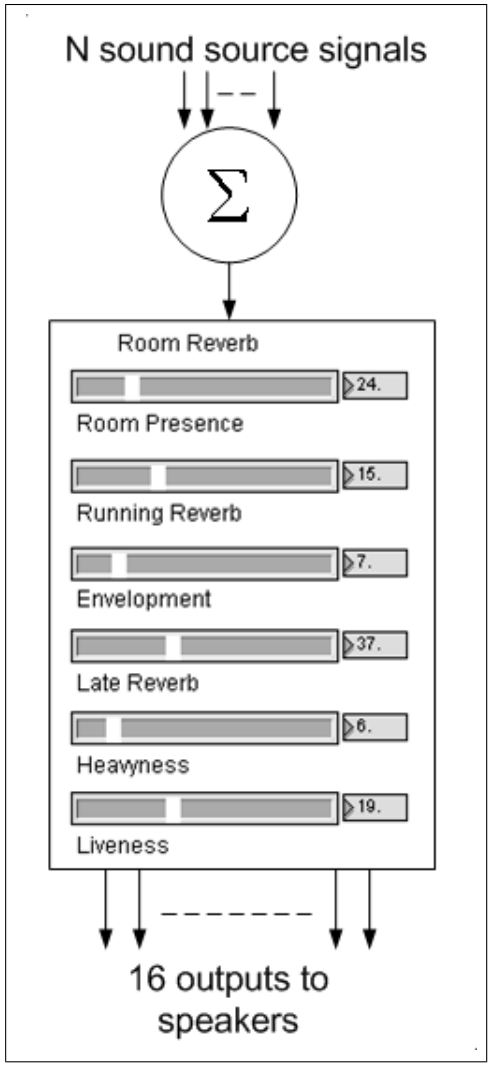


Figure 5.23: Perceptual control of room reverberation in CHESS

Having described the DSP layer of CHESS, the scene manager is now detailed.

5.5 Scene manager

The scene manager is responsible for controlling and updating the DSP layer from an XML 3D audio scene description that is based on the novel scheme developed in chapter 3. The scene manager is also used to collect user actions through a Graphical User Interface (GUI) and to provide a 3D graphical representation of the 3D audio scene being rendered. The Java programming language was selected to implement the scene manager since many tools are available in Java for parsing XML (e.g. JDOM [jdo]). Java3D was then used to visually represent scenes using simple 3D graphics. Java was also selected so that the scene manager program could be easily ported to a different platform. The overview of the scene manager is given in Fig. 5.24.

The Java scene manager program was implemented by Mark O'Dwyer[17] in his Masters thesis [O'D03] from guidelines given by the author. Currently, not all of the features of the XML 3D audio description scheme are implemented. The implemented features are: creation and control of sound sources, reflective surfaces and room reverberation. In a later stage, advanced features of the scheme such as scene score opcodes, environments and macro-objects (see chapter 3) will be implemented. However, the structure of the scene manager described here remains valid and can be used as a base to implement these advanced features.

In order to render a 3D audio scene from its XML description, objects of the scene must first be instantiated at the DSP layer. To do so, the scene manager first maps the initial structure of the scene in its memory and sends instantiating commands to the DSP layer. Then, during playback of the 3D audio scene, the scene state is updated in the scene manager memory and in turn, commands are sent to the DSP layer to reflect the updated state of the scene; this process is depicted in Fig. 5.25.

[17]Mark O'Dwyer is a student at the School of Electrical, Computer and Telecommunications Engineering (SECTE), Faculty of Informatics, University of Wollongong

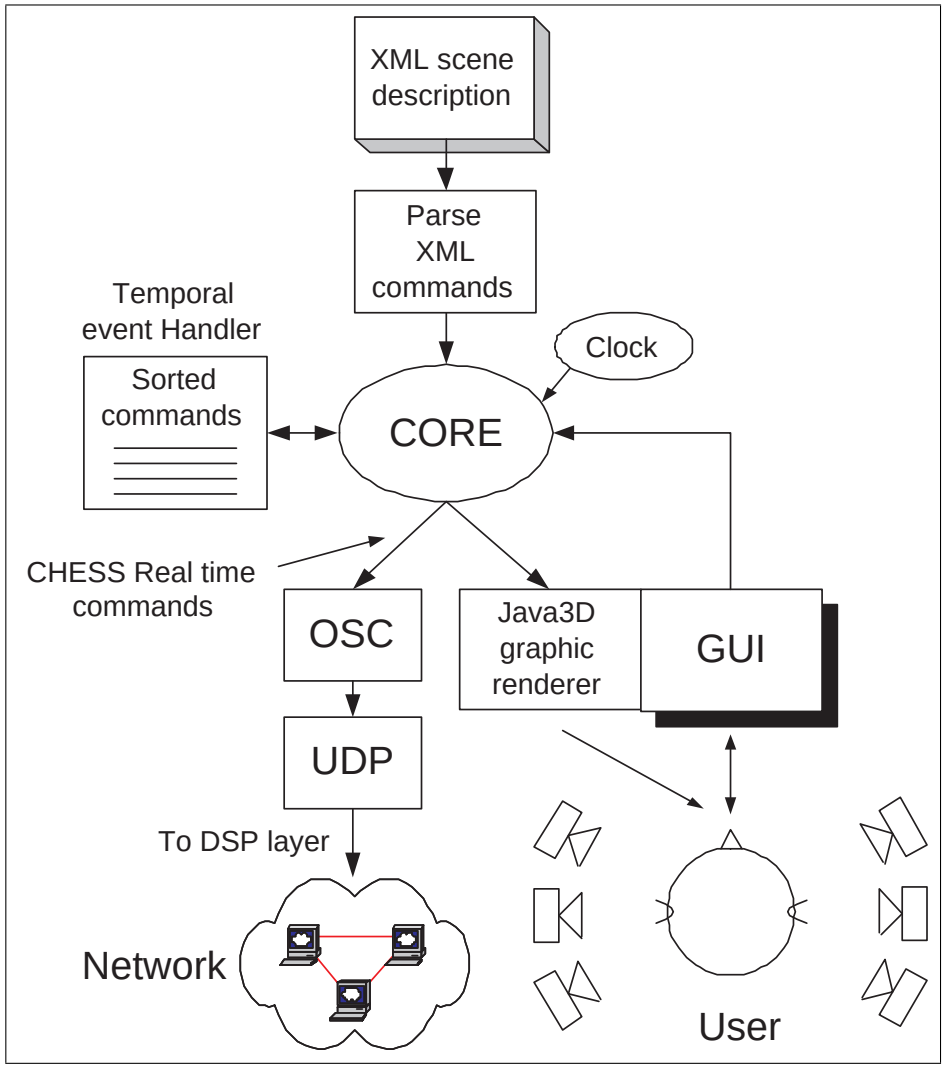


Figure 5.24: Overview of the scene manager structure

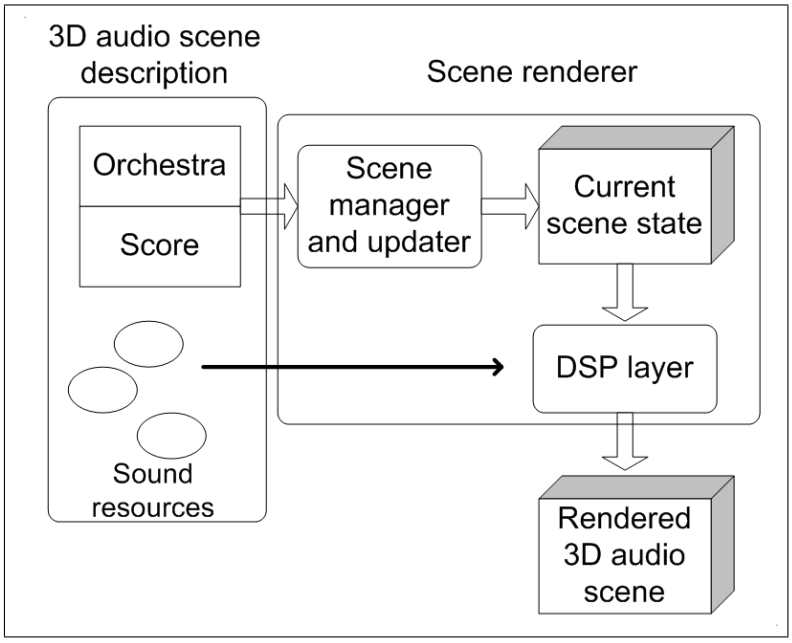


Figure 5.25: Diagram of 3D audio scene rendering from an XML scene description in CHESS

Scene initialisation

The *Core* program first parses the XML scene orchestra[18] and collects the list of objects (e.g. sound sources, surfaces, etc.) that are to be instantiated at the DSP layer. XML parsing is performed using the Document Object Model (DOM) method [dom] which is used to travel down the XML tree and to collect data. The JDOM [jdo] Java library was used to perform DOM parsing.

After parsing of the scene orchestra, instantiating commands are sent for each object present in the scene orchestra to the DSP layer via the Open Sound Control (OSC) [osc] protocol encapsulated in UDP[19] [RK] packets. The OSC protocol adopts a textual format. Commands sent to the DSP layer thus consist of text strings.

The following table gives the syntax of an instantiating command sent to the DSP

[18]The format of the XML scene orchestra was detailed in section 3.3.4
[19]User Datagram Protocol

layer to instantiate a sound source in the orchestra which ID is 'source1' and is attached to the sound sample 'beach.wav':

Command	**Object type**	**Object ID**	**Parameters**
Create	Source	Source1	beach.wav

The *Core* program then processes the XML scene initialisation score (section 3.3.5). This is used to position and set object properties before the scene is being rendered.

The table below gives the syntax of a command sent to the DSP layer to position the newly created sound source 'source1' to its initial position (Azimuth: 30 deg, Elevation: 10 deg, Distance: 1 m). The format and list of instantiating commands that are understood by the DSP layer is given in Annex 8.

Object type	**Object ID**	**Commands**	**Parameters**
Source	source1	position	30,10,1

Scene performance

The *Core* program then parses the performance score (section 3.3.5) and stores lines of scores chronologically in the *Temporal Event Handler*. At this stage, the scene is ready to be rendered, and upon user interaction, the Core program starts rendering the scene by processing each chronologically tagged line of score stored in the *Temporal Event Handler*. Each line of scores then modifies the state of the scene; this is illustrated in Fig 5.26.

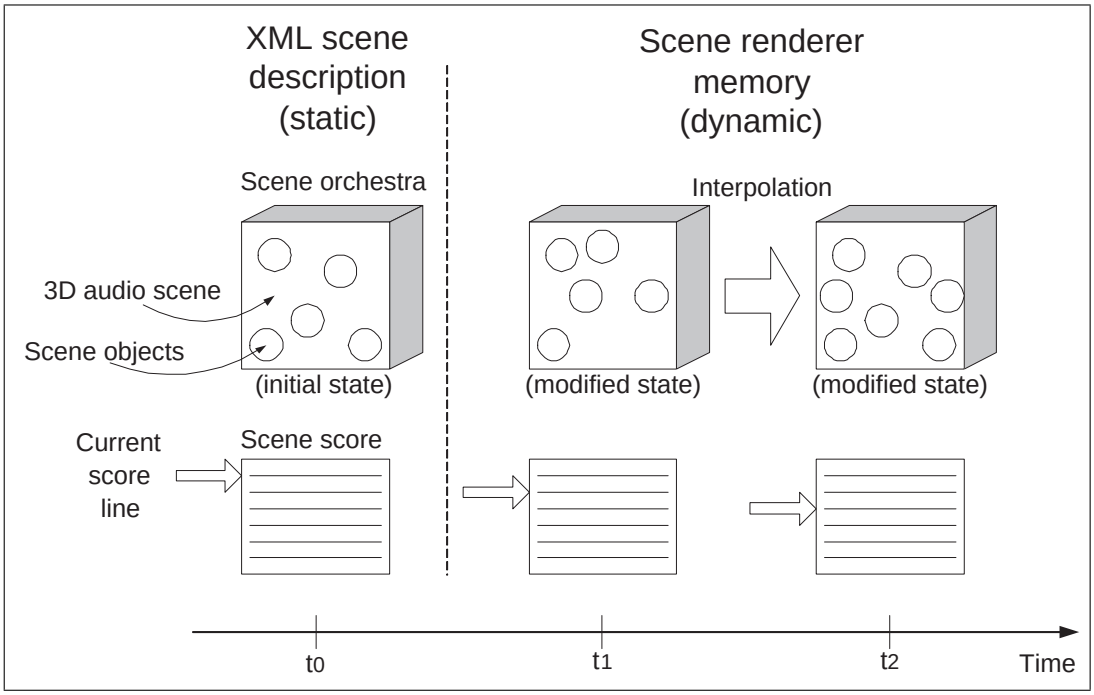


Figure 5.26: Illustration of the score modifying the current state of the orchestra in the scene renderer memory

Each line of score may result in one or multiple commands sent to the DSP layer. For example, the following line of scene score performs the displacement of a sound source to a new location:

Start time	**Duration**	**Command**	**Object(s)**	**Parameters**
0	5	Move	source1	0,5,-3

This command results in 100 scene updates commands sent to the DSP layer, since the action duration is 5 seconds and a scene update rate of 20Hz is used. The *Core* program is thus also responsible to interpolate the position of the sound source over the duration of the source displacement command defined in the scene score.

To sequence the playing times of sound sources, *play* or *stop* commands are sent to the DSP layer. For instance, the following line of scene score:

Start time	Duration	Command	Object(s)	Parameters
10	15	Play	source1	

results in a first command sent at $t = 10$ to start playing *source1* and second command at $t = 15$ to stop *source1*.

t	Object type	Object ID	Commands	Parameters
10	Source	source1	play	1
15	Source	source1	play	0

Java3D graphical rendering

The scene update commands sent to the DSP layer are also sent to the Java3D graphic renderer (Fig. 5.24). This allows sound sources and surfaces to be represented graphically. Simple or complex graphical objects may be attached to sound sources (Fig. 5.27).

Like the DSP layer, the Java3D graphical renderer is updated from the scene update commands sent by the core program (Fig. 5.24). In the *Core* program, it is possible to set different update rate variables for the 3D graphical rendering and the DSP layer. This is useful as the scene update rate in the visual domain is usually higher than in the auditory domain if smooth displacement of objects is to be achieved. Thus this feature results in fewer commands being sent to the DSP layer; this, in turn, reduces network utilisation and computational load at the DSP layer to process the incoming UDP packets.

Graphical User Interface

The system user, through a Graphical User Interface (GUI) shown in Fig. 5.27 is also able to generate instantiating and scene update commands. Currently the user can type a command string which is then sent directly to the DSP layer or he/she may modify the properties of objects (position of objects etc.) using a mouse. Initial work has begun [O'D03] on using a 3D glove, virtual reality glasses and head tracker devices to provide a more intuitive interface to control and interact with the renderer 3D audio scenes.

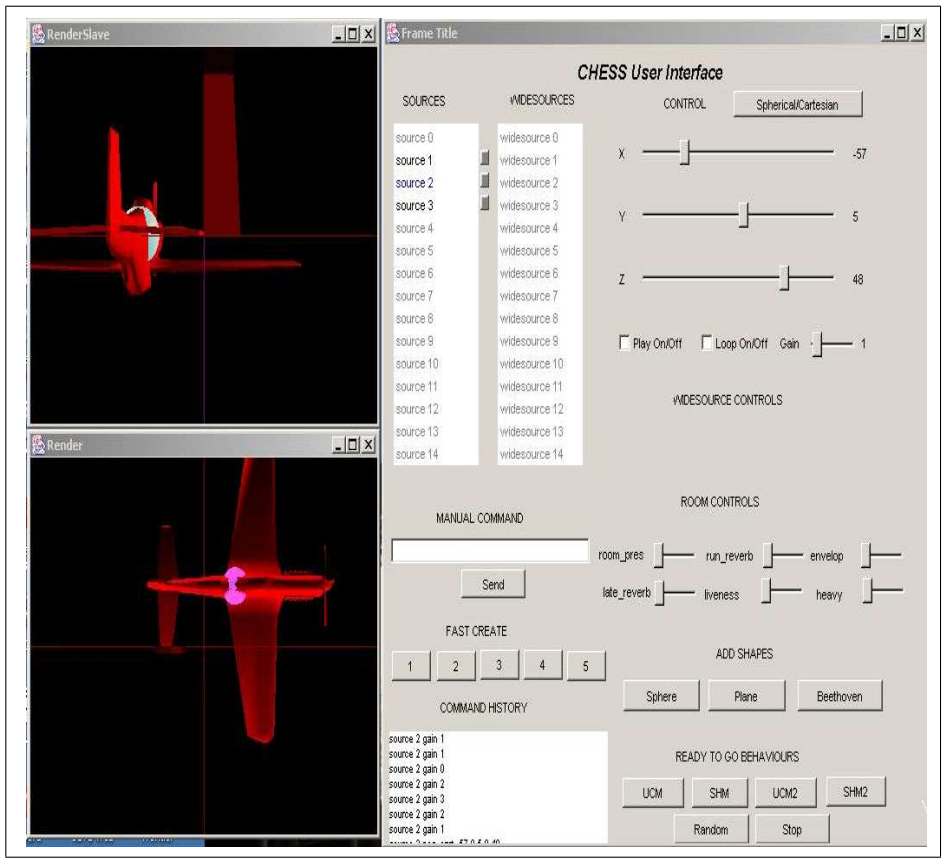


Figure 5.27: Graphical interface of the Java3D scene manager

5.6 Evaluation

5.6.1 3D audio rendering quality

Spatialisation accuracy

After implementation of the CHESS DSP layer, the precision of the 4th Order Ambisonics techniques was subjectively tested. The HOA spatialisation (section 5.4.1) technique provided a very satisfactory localisation of virtual sound sources which Azimuth and Elevation could be precisely controlled over the surface of the hemispheric region defined by the speaker array. During displacements of the sound sources on circular trajectories around the listener, there was no perceptible difference in sound source timbre, size or localisation sharpness. The position of the sound source was also stable during head rotations.

Due to the limited size of CHESS, the system is best enjoyed by one person seating in the middle of the sphere; during demonstration of the system approximately 3-4 people at the center of the array could satisfactorily experience the rendered 3D audio scenes simultaneously.

The decoding of a B-format recording (onto the 16-speaker array) captured in a highly reverberant church also gave a convincing sensation of envelopment and spaciousness.

In informal experiments, first order Ambisonics spatialisation gave, in comparison to HOA, a smeared virtual sound source, with a wider extent and higher localisation blur. It was also noticed that first order Ambisonics suffered from instabilities during head rotations and the sweet spot area only allowed one person to listen at the center of the array.

The VBAP spatialisation technique was also tested. While VBAP also provided sharp localisation of virtual sound sources, a distracting effect occurred when the algorithm switched between 2 and 3 speaker panning, depending on the position of the sound source. This effect resulted in a change in timbre and extent of the sound

source being spatialised. While VBAP was used in CHESS for certain projects (section 5.7), the use of HOA spatialisation was preferred due to its flexibility, scalability (section 5.4.1) and ability to produce hybrid scenes (section 5.3.2). Section 6.3 (further work) suggestes how different spatialisation techniques could be formally tested in psychoacoustic experiments.

Source distance

The rendering of sound source distance was explained in section 5.4.3. When room reverberation was implemented, a compelling impression of sound source distance could be rendered. However, without reverberation, the attenuation and air filtering effect cues were not sufficient to produce an impression of change in source distance. Instead, a change in sound source intensity was perceived. This is normal since the reverberant to direct sound ratio is the most important cue in determining sound source distance (2.5.2). Rendering sound source distance without relying on reverberation is still problematic, and there is room for further research to be carried out in this area.

Due to the spatialisation technique used, sound sources could not be placed in the near-field, that is between the listener and the speaker array. In the further work section 6.3 it is suggested how this issue could be addressed.

Sound source extent

The rendering of sound source extent in CHESS was described in section 5.4.4. Due to computational limitations, only three decorrelated point sources were used to render sound source extent. Although effective for sound sources with an extent smaller than 90 degrees, three point sources is insufficient for source extent wider than this value. With more computational power available at the DSP layer, more decorrelated signals could be obtained and sound sources with 2D and shape extent (chapter 4) could be devised.

Reflections

The simulation of reflections via a first image model (section 5.4.7) was used to calculate the early reverberation pattern of a room. While this required a lot of processing power since phantom sound sources had to be spatialised for each reflection, it was found that the perceptual impact was negligible when the reflections occurred shortly after the original sound source. The presence of reflections created only a change in source timbre. Simulation of reflections was however perceptible for distant reflections which could be perceived distinctly as echoes. This is due to the precedence effect that was explained in 2.9.1. Thus, it was found that the calculation of first order reflections was not perceptually important and that the high processing cost that was required was not justified.

To the author's knowledge, most real-time 3D audio systems only calculate reflections up to the second order [MW02] and it can be postulated that this is sufficient in terms of perceptual relevance. In the future, with more computational power available, acoustical prediction software (e.g. CATT [CAT]) may be ported into real-time implementations so that room reverberation could be calculated using only a purely physical approach. The efficient beam-tracing approach described by Funkhouser [FTCa04, FMC99, FCE98] could also be used.

In a future version of CHESS, diffusion effects [Emb00, KBAA01] could also be simulated.

Sound source occlusion

The detection and simulation of sound source occlusions were described in section 5.4.6. When sound source occlusion was detected, a low-pass filter with 500Hz cut-off frequency was applied on the source signal to simulate muffling. This simple approach was perceptually relevant, however it does not take into account diffraction effects. For instance, obstacles smaller than the wave length of the source signal may not have such a dramatic effect since the source signal is allowed to bend around the obstacle. Therefore the technique that was described in section 5.4.6 and which uses a light

beam approach to detect occlusion should be used only for large obstacles (relative to the sound source largest wavelength). Although not implemented in CHESS for performance reasons, the size of the obstructing surface could be compared with the mean wavelength of the source signal to determine the amount of low-pass filtering to be applied on the occluded sound source signal; so that more low-pass/muffling is applied when the dimension of the obstacle is greater than the wavelength of the source signal.

A solution to the simulation of diffraction effects is also given in [HSHT96]. This approach consists of simulating diffraction via an edge sound source representing the bending of the source signal around the obstacle. Since low frequencies of the source signal are more prone to diffraction, a frequency filtering should be applied on the edge source. Although interesting and a possible candidate for emulating diffraction in later versions of CHESS, this method has not been subjectively tested.

Delays and Doppler effect

The simulation of delays and the Doppler effect was described in section 5.4.5. Pitch bends of the sound source signal was very effective in rendering sound source movement. However, during very fast movements of sound sources the Doppler effect should be clamped as it can produce excessive pitch bends which can sound unnatural.

Reverberation

Room reverberation was described in section 5.4.8. Room reverberation simulation followed a Feedback Delay Network implementation controlled by perceptual parameters. By altering the parameters, a wide range of acoustical environments could be created. However, if precise acoustics is required, this approach provides only a subjective estimation. CHESS could be extended by using convolution [FT98] of the source signal with a B-format impulse response captured in the acoustical space that

is to be emulated. The resulting B-format would then be added on the 'Ambisonics bus' and decoded along with the sound source signals.

5.6.2 System structure

The client-server approach of CHESS was described in section 5.2. This approach allows the DSP layer that performs all the processing tasks to be controlled remotely. This approach also allowed the separation of the scene management tasks and 3D graphics rendering from the 3D audio rendering tasks. The latency of the whole system (i.e. the time it takes between an user action on the client and when it is reflected at the DSP layer) was estimated to reach approximatively 40ms and was not perceptually noticeable.

The OSC and UDP protocols were reliable in establishing a connection between the client and the server. The UDP network protocol, unlike TCP/IP, does not require to establish a connection before sending packets and does not send acknowledgement packets. This simpler behaviour permits lower computational load at the DSP layer to process the control packets emitted by the scene manager. It was found that packet loss was negligible except in extreme cases where the scene manager sent too many packets because of the scene refresh rate being set to a high value (around 200 Hz). In the CHESS system, the physical distance between the scene manager and the DSP layer was however short (i.e. on the same network) and thus, it would interesting to carry out a test where the DSP layer is controlled by the scene manager from a remote location (e.g. in another country).

Speaker array

The speaker array described in section 5.2.2 provided a useful tool for quickly changing speaker configurations. To our knowledge only one other 3D audio rendering system described in [KKFW99] uses a configurable speaker arrangement. The size of the array was sometimes problematic during events and demonstrations when a large audience

(10 people) was inside the array. However, CHESS should be seen as a reduced scale of a larger 3D audio rendering environment; such as the one implemented using the CHESS equipment in section 5.29.

Due to the limitation on the number of the sound card output channels, a sub-woofer could not be used when all 16 speakers were used. Therefore in later versions of CHESS more output channels will be available. The number of speakers could also be doubled so that a fully spherical 3D audio rendering could be implemented.

The acoustics of the room where CHESS was built was acoustically treated but it could be improved by covering walls and windows with absorbing foam. The impact of the natural environment was acceptable, however by cancelling problematic reflections, even better spatialisation accuracy could be achieved.

The DSP layer

The DSP layer was implemented in the Max/Msp language (section 5.3) and several objects were programmed in C when higher efficiency was required. In comparison to a textual programming language (e.g. C++) Max/Msp allowed very fast system prototyping. A drawback with Max/Msp is that a 'click' occurs when the DSP chain is broken or changed (for example when a new sound source is instantiated during rendering), this is a problem currently inherent to the Max/Msp platform which may be fixed in later versions.

With the processing speed of the computer on which the DSP layer was implemented (Macintosh G4 867MHz) 24 sound sources could be spatialised in real-time before reaching the computational limit. Despite being a large number of simultaneous sound sources in a 3D audio scene, this number is not so high when surfaces (creating phantom sources) are used in the scene.

Scene manager

The scene manager implemented in Java and Java3D by Mark O'Dwyer, from guidelines given by the author, demonstrated that the CHESS DSP layer could be controlled remotely from any computer connected on the internet. The scene manager could also parse 3D audio scenes described in XML and then controls the DSP layer accordingly. At present, XML scenes have to be hand coded; in a later stage the Java3D scene manager will be developed as a scene authoring tool to compose 3D audio scenes and save them in XML syntax using the novel scheme described in chapter 3. The XML scheme provided a straight forward approach to developing the scene manager program. The scene score commands were easily converted to DSP layer commands. Using MPEG-4 AudioBIFS or VRML instead of the proposed scheme would have lead to a much greater complexity of the scene manager program.

5.7 Practical uses of CHESS

Several applications where CHESS was used are now briefly described.

Psychoacoustic experiments

CHESS was used to perform the psychoacoustic experiments on sound source extent that are described in chapter 4. CHESS proved to be a useful tool, since it allowed various speaker configurations to be quickly arranged.

Listening to the mind listening

CHESS was used in the project entitled "Listening to the Mind Listening" [LML04] which was the world's first sonification[20] concert which took place at the Sydney Opera House on the 8th of July 2004. The concert consisted of several sonification

[20]Sonification is the process of translating data into sound

pieces of EEG data of a person's brain listening to a piece music, which was only revealed at the end of the concert.

CHESS was used as the main processing station for concert submissions and to spatialise sonified pieces onto a custom 15-speaker array using VBAP spatialisation [Pul97]; Ambisonics could not be used due to the irregularity of the speaker array. As part of this project, CHESS was also use to create the piece entitled "Neural Dialogues" [PS04] which was one of the ten accepted pieces played during the concert at the Opera House. This piece was created by Greg Schiemer[21] and Guillaume Potard directly on the CHESS system. CHESS proved to be a useful system since the flexibility of the speaker array allowed a scaled down version of the speaker array used at the Sydney Opera House (details in [LML04]) to be reproduced.

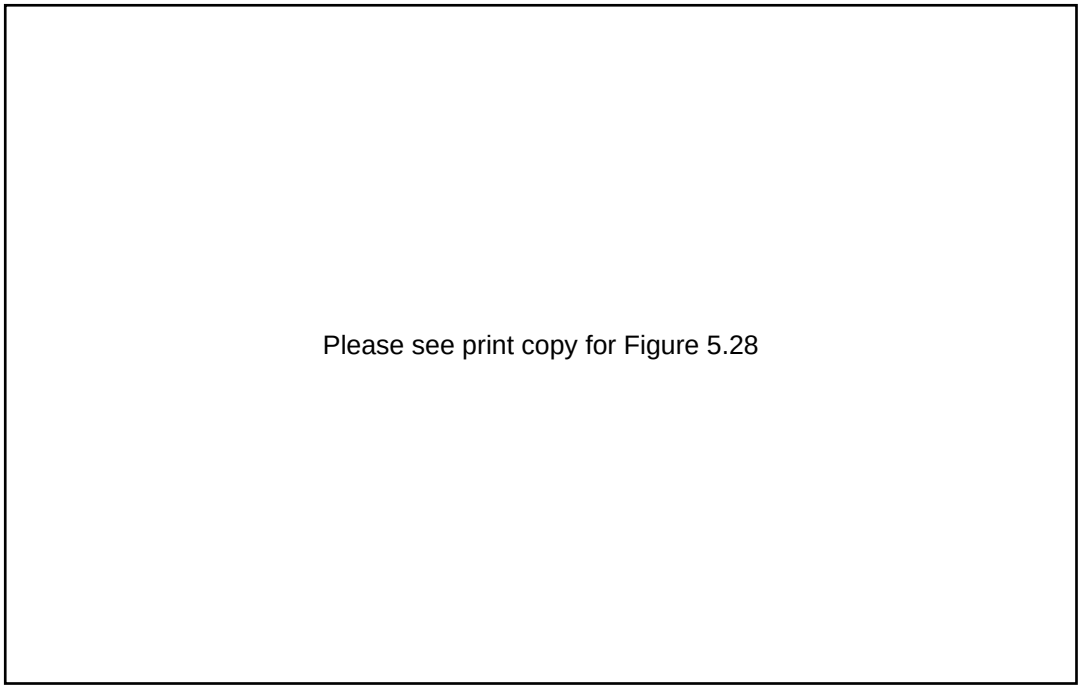


Figure 5.28: Listening to the mind listening promotion

[21] Greg Schiemer is associate Professor at the Faculty of Creative Arts, University of Wollongong

Sonic Connections

Sonic connections [son] is an annual academic music festival organised by Greg Schiemer at the Faculty of Creative Arts, University of Wollongong. At Sonic Connections 2004, CHESS was used in an outdoor context by placing the 16-speaker array in a portable geodesic dome structure provided by David Worrall (Fig. 5.29). This allowed testing of CHESS on a larger scale in which 50 people could simultaneously experience 3D audio compositions being played.

The 4th order Ambisonics spatialisation plugins described in 5.4.2 were used by Dwight Mowbray and Brent Williams[22] to compose the piece "Sikanex" that was played at Sonic Connections. This demonstrated the useability of these new spatialisation plugins.

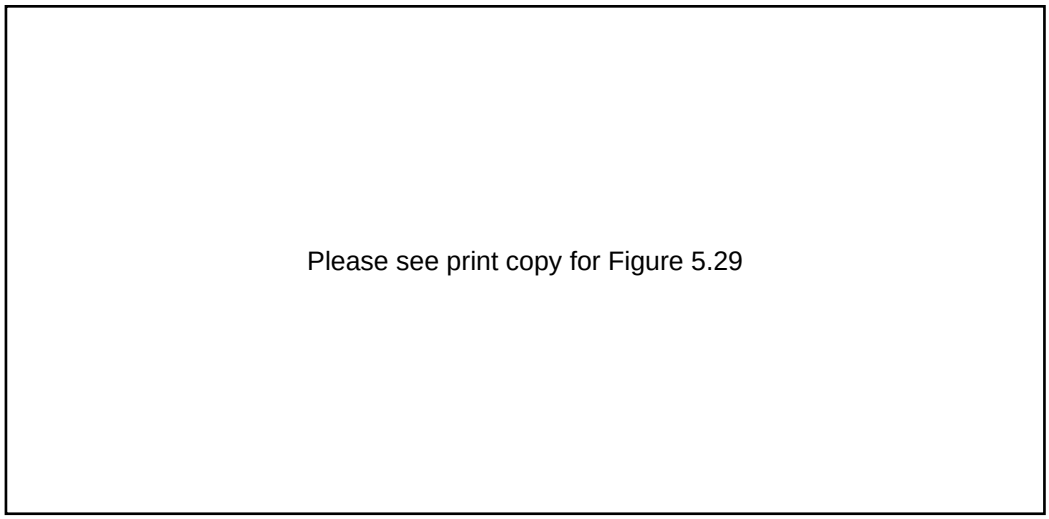


Figure 5.29: Picture of the outdoor CHESS dome at Sonic Connections 2004

[22]Students at the Faculty of Creative arts, University of Wollongong

5.8 Summary

In this chapter, the novel Configurable Hemispheric Environment for Spatialised Sound (CHESS) system was described. The CHESS system provides a versatile environment for rendering 3D audio scenes that can be experienced simultaneously by several users. The architecture and techniques used in CHESS can be used as a model to implement subsequent 3D audio rendering systems.

The CHESS system consists of two parts: a DSP layer implementing all the DSP tasks necessary to render 3D audio scenes and a scene manager acting as a remote control, XML scene parser and a 3D graphical user interface. The implementation of the CHESS system also showed that rendering of 3D audio scenes from the XML 3D audio scene description scheme described in chapter 3 was viable.

The advantages of the Ambisonics spatialisation used in CHESS were also highlighted. These include the ability to produce 3D audio content that can be later decoded on different speaker configurations and the ability to create hybrid 3D audio scenes. The different processing tasks of the DSP layer were also detailed; these include rendering of source distance, source extent, occlusions, reflections and reverberation. The reasoning and justification behind the choice of certain 3D audio rendering techniques over others was discussed.

The research and creative potential of the CHESS system is clearly immense and to this date, only a fraction of this potential has been explored in creative projects.

Chapter 6

Conclusions and further work

6.1 3D audio scene description

A novel scheme named XML3DAUDIO for describing 3D audio scenes in an object oriented way was proposed in chapter 3 and its implementation in a 3D audio rendering system was detailed in chapter 5. While having 3D audio description capabilities that are superior to that of MPEG-4 AudioBIFS, 3D audio scenes that are described using XML3DAUDIO have a much simpler syntactic structure. Scenes described with XML3DAUDIO are human readable and they can be easily authored or modified using a simple text or XML editor.

XML3DAUDIO follows the new scene orchestra and score approach which permits separating the scene content data (i.e. the objects present in the scene) from the scene temporal data (i.e. sequencing of sound sources and object animation). This approach also allows, by using special commands in the scene score, to devise 3D audio scene algorithmically. A short list of scene composition commands was given and it was shown how XML3DAUDIO could be extended with more complex scene composition commands. Complex scene interactivity features such as found in VRML

or MPEG-4 BIFS are not implemented in the current state of XML3DAUDIO, since when rendering 3D audio only scenes, complex scene interactivity behaviours are usually not necessary. However, if these interactivity behaviours are needed, they could be described by defining new commands in the scene score.

One difference between XML3DAUDIO and MPEG-4 AudioBIFS is that XML3DAUDIO only describes the scene and thus, sound resources are external to the scene, in contrast, MPEG-4 BIFS provides a binary framework where the scene description and the sound resources can be multiplexed into a single binary stream. A solution to this problem which has been considered [PB02b] is to combine XML3DAUDIO with the MPEG-21 standard [BVdWH+03]. This technique would allow merging an XML3DAUDIO scene description and its related sound resources into a single MPEG-21 Digital Item. This wrapping mechanism would thus allow 3D audio scenes to be transmitted as single entities. In addition, MPEG-21 Digital Item Adaptation could be employed to perform adaptation of 3D audio content, for instance, to compensate the acoustics of the user environment, or to render 3D audio scenes using the user's own HRTF. Use of MPEG-21 to perform adaption of 3D audio content was proposed by the author in [PB02c].

6.2 Sound source extent and shape

A technique was reviewed in section 2.12, whereby spatialising several several decorrelated sound sources, the apparent extent and shape of sound sources could be controlled. This technique is based on the ability of the brain to merge several sound sources that are perceptually similar into a single auditory stream. Decorrelation is used so that the summing localisation effect is defeated; this would otherwise result in a narrow apparent extent located at the center of gravity of the sound sources.

This technique was first tested to create horizontally extended sound sources; it

was found that the mean perceived extent matched fairly closely the source extent that was intended. However, when sound source density was too high, this narrowed the perceived extent; it was suggested that this effect was due to an increase of the IACC coefficient. On the other hand, when sound sources were too sparsely distributed, subjects could perceive individual sound sources instead of a single extended sound source; binaural fusion was lost. It was found that a density of one source per 30 degrees was a good compromise between these two extremes.

Two experiments then studied the ability by subjects to identify the apparent shape of sound sources. Results showed that correct identification of sound source shape by subjects was above statistical probability but always less than 50 % of the time. It was suggested that this could be due to errors introduced by the experiment setup, since the spatialised decorrelated sound sources used to create the sound shapes were not punctual, and their minimal size was defined by the size of the speakers used. To further the understanding of sound source shape perception by listeners, it would be interesting to carry experiments where the sound source shape stimuli are created using natural sound sources which emit sound continuously over a surface or volume (e.g. a vibrating pannel or a three-dimensional array of spatially small sound sources).

It was explained that, when decorrelation was applied to signals having time varying spectra, the amount of decorrelation varied with the signal spectrum. Therefore, for these signals, source extent varied with the energy of the signal spectrum and this can explain that poorer results of correct sound source shape identification were obtained when a music signal was used. The author intends in the future to devise new decorrelation techniques that take into account the nature of the input signal so as to provide a fixed amount of decorrelation between signals which is independent of the spectral variations of the input signal; this would allow controlling the extent and shape of sound sources for any type of signal.

In the experiments, only one and two dimensional extended sound sources were devised and tested, however, some natural sound sources such as a swarm of insect of wind blowing in trees may have a three-dimensional extent, that is, the depth of the sound source may also be perceived. To carry experiments studying the perception of 3D extended sound sources, a spatialisation technique which allows a precise control of sound source distance would need to be used. Currently, the Wave Field Synthesis (WFS) spatialisation technique is a good candidate to perform this research since it provides accurate rendering of sound source distance.

The effects of subject training were not investigated. However, since the perception of sound shape is not an usual conscious action, it can be expected that subject training would likely improve their ability to perceive sound source shapes.

Effects of visions on sound source extent perception were also not studied. In real life, it is expected that humans use several inter-modal cues to determine the size of sound emitting objects. Some of the experiments presented in this chapter could be thus be carried out with and without visual masking or with blind subjects as to study this effect.

6.3 3D audio rendering

The practical implementation of a novel 3D audio rendering system was detailed in section 5. The system known as CHESS can, from an XML scene description, render 3D audio scenes to a small audience. CHESS can also be used in real-time or within a sequencer program using new spatialisation software plug-ins that were presented. CHESS is composed of two main modules: the scene manager and the DSP layer. The scene manager was implemented in Java and Java3D. Currently, the scene manager is used to render XML scene descriptions and to control the DSP layer accordingly,

which performs all 3D audio processing tasks. The scene manager Java3D implementation could be further improved by developing a full 3D audio scene authoring suite which then saves scenes in the XML3DAUDIO format. Initial work begun on using a 3D glove and virtual reality glasses, these interfaces would allow a more intuitive interaction with the 3D audio scenes than the current graphical user interface which uses a mouse.

The rendering quality of the 4th order Ambisonics spatialisation technique used in CHESS proved to be good and allowed sharp localisation of sound sources. However, while it is possible to create the illusion of far away sound sources, sound sources cannot be placed inside the speaker array, which could dramatically improve the realism of the rendered 3D audio scenes. To produce such near-field sound sources, the Ambisonics technique that is currently used could be improved with a recent development of Ambisonics which is called distance coding [Dan03b]. This technique allows (in theory) the placing of sound sources inside the speaker array; to the knowledge of the author a subjective evaluation of this technique has not been published. Another option for placing sound sources within the speaker array would be to use an Audio Spotlight technique [spo] which, by using ultrasound beams, can produce very sharp virtual sound sources in space.

6.4 General conclusion

This thesis presented contributions, novel techniques and new findings in three interwoven domains of 3D audio:

- 3D audio scene description
- 3D audio rendering techniques
- 3D audio perception

These three areas cover the three steps that are involved in delivering 3D audio content to a listener: encoding-rendering-perception.

This thesis showed that the object oriented approach was beneficial when encoding and rendering 3D audio scenes. In the end, however, subjective perception and psychoacoustics remain the only judges to determine if 3D audio content is indeed being delivered successfully.

Bibliography

[AB79] J.B. Allen and D.A. Berkley. Image method for efficiently simulating small-room acoustics. *Journal of the Acoustical Society of America*, 65(4):943–950, 1979.

[ADT04] V. R. Algazi, R.O. Duda, and M. Thompson, D. Motion-tracked binaural sound. In *116th Audio Engineering Convention*, Berlin, Germany, 2004.

[AEH+00] O. Avaro, A. Eleftheriadis, C. Herpel, G. Rajan, and L. Ward. Mpeg-4 systems: Overview. *Signal Processing: Image Communication*, 15:281–298, 2000.

[Ali98] M. Ali. Stereophonic acoustic echo cancellation system using time-varying all-pass filtering for signal decorrelation. In *ICASSP 98*, 1998.

[ANM97] A. L. Ames, D.R. Nadeau, and J.L. Moreland. *VRML 2.0 Sourcebook*. John Wiley and Sons, 1997.

[Aro00] B. Arons. A review of the cocktail party effect, 2000.

[Baa03] M.A.J. Baalman. Application of wave field synthesis in the composition of electronic music. In *ICMC 2003*, Singapore, 2003.

[Bar99] M. Barron. Spatial impression and envelopment in concert halls. *Proceedings of the Institute of Acoustics*, 21(6):163–169, 1999.

[Bat67] D. W. Batteau. The role of the pinna in human sound localization. *Proceedings of Royal Society*, 168:158–180, 1967.

[Bau61] B. Bauer. Phasor analysis of some stereophonic phenomenon. *Journal of the Acoustical Society of America*, 33:1336–1539, 1961.

[Bau63] B. Bauer. Some techniques toward better stereophonic perspective. *IEEE Transactions on Audio*, AU-11(88), 1963.

[Bau93] J. Bauck. Developments in transaural stereo. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 1993.

[BB72] H.E. Bass and H.-J. Bauer. Atmospheric absorption of sound: Analytical expressions. *Journal of the Acoustical Society of America*, 3(2), 1972.

[Beg91a] D.R. Begault. Challenges to the successful implementation of 3-d sound. *Journal of the Audio Engineering Society*, 39(11):864–870, 1991.

[Beg91b] D.R. Begault. Preferred sound intensity increase for sensation of half distance. *Perceptual and motor skills*, 72:1019–1029, 1991.

[Beg92a] D.R. Begault. Binaural auralization and perceptual veridicality. In *93rd AES convention*, San Francisco, 1992.

[Beg92b] D.R. Begault. Perceptual effects of synthetic reverberation on three-dimensional audio systems. *Journal of the Audio Engineering Society*, 40(11):895–904, 1992.

[Beg94] D.R. Begault. *3-D Sound for virtual reality and Multimedia*. Academic Press, 1994.

[Bek62] von Bekesy. Hearing theories and complex sounds. *Journal of the Acoustical Society of America*, 35(4):588–601, 1962.

[Ber88] A.J. Berkout. A holographic approach to acoustic control. *Journal of the Audio Engineering Society*, 36(12):977–995, 1988.

[Ber96] L. Beranek. *Concert and Opera Halls: How they Sound*. Acoustical Society of America, 1996.

[BGP03] J. Bormans, J. Gelissen, and A. Perki. Mpeg-21: The 21st century multimedia framework. *IEEE Signal Processing Magazine*, 20(2):53 – 62, March 2003 2003.

[BL85] J. Blauert and W. Lindermann. Spatial mapping of intracranial auditory events for various degrees of interaural coherence. *Journal of the Acoustical Society of America*, 79(3):806–813, 1985.

[BL86] J. Blauert and W. Lindermann. Auditory spaciousness: Some further psychoacoustic analyses. *Journal of the Acoustical Society of America*, 90(2):533–542, 1986.

[Bla97] J. Blauert. *Spatial hearing : the psychophysics of human sound localization*. MIT Press, Cambridge, Mass., rev. edition, 1997.

[Bla02] J. Blauert. Hearing of music in three spatial dimensions, 2002.

[Blu33] A. D. Blumlein. Improvements in and relating to sound-reproduction systems, british patent 394,325, 1933.

[Boo95] M.M. Boone. Spatial sound-field reproducion by wave-field synthesis. *Journal of the Audio Engineering Society*, 43(12):1003–1011, 1995.

[Bor26] E.G. Boring. Auditory theory with special reference to intensity, volume and localization. *Journal of Psychology*, 37(2):157–188, 1926.

[Bor84] J. Borish. Extension of the image model to arbitrary polyhedra. *Journal of the Acoustical Society of America*, 75(6):1827–1836, 1984.

[Bos00] M. Bosi. Multichannel audio coding and its applications in dab and dvb. In *ICSP 2000*, 2000.

[Bou00] R. C. Boulanger. *The Csound book : perspectives in software synthesis, sound design, signal processing, and programming.* MIT Press, Cambridge, Mass., 2000.

[Bre94] A.S. Bregman. *Auditory scene analysis: the perceptual organization of sound.* MIT Press, Cambridge, 1994.

[BSZ95] H.E. Bass, L.C. Sutherland, and A.J. Zuckerwar. Atmospheric absorption of sound: Further developments. *Journal of the Acoustical Society of America*, 97(1):680–683, 1995.

[BVdWH+03] I. Burnett, R. Van de Walle, K. Hill, J. Bormans, and F. Pereira. Mpeg-21: Goals and achievements. *IEEE Multimedia magazine*, 10(4):60–70, October-December 2003 2003.

[Cab02] D. Cabrera. The size of sound: Auditory volume reassessed, 2002.

[Cah01] M. Cahill. Using xml for score representation. In *DAFX 2001*, Limerick, Ireland, 2001.

[Car96] S. Carlile. *Virtual Auditory Space: Generation and Application.* R.G. Landes Company, Austin, TX, USA, 1996.

[CAT] Catt acoustics, http://www.catt.se/.

[CD68] R.I. Chernyak and N.A. Dubrovsky. Pattern of the noise images and the binaural summation of loudness for the different interaural correlation of noise. In *6th International Congress on Acoustics*, pages 53–56, Tokyo, Japan, 1968.

[CETT02] P. Cook, G. Essl, G. Tzanetakis, and D. Trueman. N ¿¿ 2: Multi-speaker display systems for virtual reality and spatial audio projection.

In *8th International Conference on Auditory Displays*, Kyoto, Japan, 2002.

[Che53] E. C. Cherry. Some experiments on the recognition of speech, with one and with two ears. *Journal of the Acoustical Society of America*, 25(5):975–979, 1953.

[Cho71] J. Chowning. The simulation of moving sound sources. *Journal of the Audio Engineering Society*, 19(1):2–6, 1971.

[CPCS05] M. Chait, D. Poeppel, A. Cheveigne, and J. Simon. Human auditory cortical processing of changes in interaural correlation. *Journal of Neuroscience*, 25(37):8518–8527, 2005.

[Dam68] P. Damaske. Subjektive untersuchung vob schallfeldern. *Acustica*, 19:199–213, 1967/1968.

[Dan00] J. Daniel. *Reprsentation de champs acoustiques, application la transmission et la reproduction de scnes sonores complexes dans un contexte multimdia*. Phd, Universit Paris 6, 2000.

[Dan03a] J. Daniel. Further investigations of high order ambisonics and wavefield synthesis for holophonic sound imaging. In *114th Audio Engineering Convention*, Amsterdam, The Netherlands, 2003.

[Dan03b] J. Daniel. Spatial sound encoding including near field effect: Introducing distance coding filters and a viable, new ambisonic format. In *AES 23rd International Conference*, Copenhagen, Denmark, 2003.

[Dav03] M.F. Davis. History of spatial coding. *Journal of the Acoustical Society of America*, 51(6):554–569, 2003.

[DFMM98] G. Dickins, M. Flax, A. McKeag, and D. McGrath. Optimal 3d speaker panning. In *AES 116th International Conference*, 1998.

[Dol98] Dolby. Dolby digital - the sound of the future here today, 1998.

[dom] W3c document object model (dom), www.w3.org/dom/.

[DRP98] J. Daniel, J.-B. Rault, and J.-D. Polach. Ambisonics encoding of other formats for multiple listening conditions, aes preprint no. 4795,. In *Audio Engineering Society 105th Convention*, San Francisco, USA, 1998.

[DRS+03] A. Dantele, U. Reiter, M. Schuldt, H. Drumm, and O. Baum. Implementation of mpeg-4 audio nodes in an interactive virtual 3d environment. In *Audio Engineering Society 114th Convention*, Amsterdam, The Netherlands, 2003.

[Dru97] I. A. Drumm. *The Development and Application of an Adaptive Beam Tracing Algorithm to Predict the Acoustics of Auditoria*. Phd, University of Salford, 1997.

[EAX] EAX. Creative technology ltd., environmental audio extensions: Eax 2.0, eax 2.0 extensions sdk, http://developer.creative.com/.

[Ell98] S. Ellis. Towards more realistic sound in vrml. In *VRML 98 : Third Symposium on the Virtual Reality Modeling Language*, Monterey, California,, 1998.

[Emb00] J.J. Embrechts. Broad spectrum diffusion model for room acoustics ray-tracing algorithms. *Journal of the Acoustical Society of America*, 107(4):2068–2081, 2000.

[Evr01] G. Evreinov. Spotty: Imaging sonification based on spot-mapping and tonal volume. In *ICAD 2001*, Espoo, Finland, 2001.

[Fal04] C. Faller. Binaural cue coding: Rendering of sources mixed into a mono signal. In *DAFX 2004*, Naples, Italy, 2004.

[FBAA03] H. Farag, J. Blauert, and O. Abdel Alim. Psychoacoustic investigations on sound-source occlusion. *Journal of the Audio Engineering Society*, 51(7/8):635–646, 2003.

[FCE98] T. Funkhouser, I. Carlbom, and G. Elko. A beam tracing approach to acoustic modeling for interactive virtual environments. In *Siggraph 98*, pages 21–32, 1998.

[FM] R. Furse and D.G. Malham. First and second order ambisonic decoding equations, www.muse.demon.co.uk/ref/speakers.html.

[FMC99] T. Funkhouser, P. Min, and I. Carlbom. Real-time acoustic modeling for distributed virtual environments. In *ACM Computer Graphics, Siggraph 99*, pages 365–374, 1999.

[FP02] C. Fancourt and L. Parra. A comparison of decorrelation criteria for the blind source separation of nonstationary signals. In *IEEE Sensor Array and Multichannel signal processing workshop*, pages 165–168, Rosslyn, USA, 2002.

[FPST03] M. Foco, P. Polotti, A. Sarti, and S Tubaro. Sound spatialization based on fast feam tracing in the dual space. In *DAFX 2003*, London, UK, 2003.

[FRO02] F. Fontana, D. Rocchesso, and L. Ottaviani. A structural approach to distance rendering in personal auditory displays. In *Fourth IEEE International Conference on Multimodal Interfaces*, pages 33 – 38, 2002.

[FT98] A. Farina and L. Tronchin. 3d impulse response measurements on s.maria del fiore church, florence, italy. In *International Conference on Acoustics, Seattle (WA), 26-30 June 1998*, 1998.

[FTCa04] T. Funkhouser, N. Tsingos, I. Carlbom, and et al. A beam tracing method for interactive architectural acoustics. *Journal of the Acoustical Society of America*, 115:739–756, 2004.

[Gas03] R. Gaspal. Surround ambiophonic recording and reproduction. In *AES 25th International Conference on Multichannel Audio*, Banff, Canada, 2003.

[Ger75] M.A. Gerzon. Ambisonics: Part two: Studio techniques. *Studio Sound*, August 1975:24–30, 1975.

[Ger85] M.A. Gerzon. Ambisonics in multichannel broadcasting and video. *Journal of the Audio Engineering Society*, 33(11):859–871, 1985.

[Ger92a] M. A. Gerzon. Ambisonic decoders for hdtv. In *92nd AES Convention*, Vienna, Austria, 1992.

[Ger92b] M. A. Gerzon. General metatheory of auditory localisation. In *92nd AES convention*, Vienna, Austria, 1992.

[Ger92c] M. A. Gerzon. Hierarchical system of surround sound transmission for hdtv. In *92nd AES convention*, Vienna, Austria, 1992.

[Ger92d] M.A. Gerzon. The design of ditance panpots, preprint 3308. In *Audio Engineering Society 92nd Convention*, 1992.

[Ger92e] M.A. Gerzon. Panpot laws for multispeaker stereo. In *92nd Audio Engineering Society Convention*, Vienna, Austria, 1992.

[Ger98a] M.A. Gerzon. Decoderd for feeding irregular loudpspeaker arrays, us patent 4,414,430, 1998.

[Ger98b] M.A. Gerzon. Surround sound apparatus, us patent 5,757,927, 1998.

[Gir96] F. Giron. *Investigations about the Directivity of Sound Sources*. PhD thesis, Ruhr University, 1996.

[Gri96] D. Griesinger. Spaciousness and envelopment in musical acoustics. In *101st AES convention*, volume Preprint 4403, 1996.

[Gri97] D. Griesinger. The psychocoustics of apparent source width, spaciousness and envelopment in performace spaces. *Acustica*, 83(4):721–731, 1997.

[GS64] M. Guirao and S. S. Stevens. Measurement of auditory density. *Journal of the Acoustical Society of America*, 36(6):1176–1182, 1964.

[GT02] A. Gogu and M. Topa. Coefficients' computation for jot's reverberation algorithm. In *10th Mediterranean Electrotechnical Conference (MEleCon 2000)*, 2002.

[Har66] C.M. Harris. Absorption of sound in air versus humidity and temperature. *Journal of the Acoustical Society of America*, 40(1):148–159, 1966.

[Har83] W.M. Hartmann. Localization of sound in rooms. *Journal of the Acoustical Society of America*, 74(5), 1983.

[HDM03] H. Hoffmann, R. Dachselt, and K. Meissner. An independent declarative 3d audio format on the basis of xml. In *International Conference on Auditory Display (ICAD 2003)*, Boston, MA, USA, 2003.

[Hol94] J. Hollander. *An Exploration of Virtual Auditory Shape Perception.* Ma, University of Washington, 1994.

[How02] D. C. Howell. *Statistical methods for psychology, Fifth edition.* Duxbury Thomson Learning, 2002.

[HSHT96] J. Huopaniemi, L. Savioja, T. Huotilainen, and T. Takala. Implementation of a virtual audio reality system. In *Nordic Acoustical Meeting*, Helsinki, Finland, 1996.

[HSK97] J. Huopaniemi, L. Savioja, and M. Karjalainen. Modeling of reflections and air absorption in acoustical spaces - a digital filter design approach. In *IEEE 1997 Workshop on Applications of Signal Processing toAudio and Acoustics, New Paltz, New York, Oct. 19-22, 1997*, 1997.

[HST96] J. Huopaniemi, L. Savioja, and T. Takala. Diva virtual audio reality system. In *International Conference on Auditory Display (ICAD 96)*, Xerox PARC, California, USA, 1996.

[hur] Lake huron, audio workstation, lake technology limited, http://www.lake.com.au/driver.asp?page=main/products/huron/huron.

[ISO97] ISO-3382. Acoustics - measurement of the reverberation time of rooms with reference to other acoustical parameters, 1997 1997.

[ITU94] ITU. Itu-r recommendation bs.775-1, "multichannel stereophonic sound system with and without accompanying picture", international telecommunication union, geneva, switzerland. Technical report, 1992-1994.

[jdo] Jdom java library, www.jdom.org/.

[Jef47] L.A. Jeffress. A place theory of sound localization. *Journal of Comp. Physiol. Psych.*, 41:35–39, 1947.

[JLP99] J.M. Jot, V. Larcher, and J.-M. Pernaux. A comparative study of 3-d audio encoding and rendering techniques. In *19th Audio Engineering Society Conference*, pages 281–300, Rovaniemi, Finland, 1999.

[JLW95] J.M. Jot, V. Larcher, and O. Warusfel. Digital signal processing issues in the context of binaural and transaural stereophony. In *98th AES Convention*, Paris, 1995.

[Jot96] J.M. Jot. Synthesizing three-dimensional sound scenes in audio or multimedia production and interactive human-computer interfaces. In *5th International conference: Interface to Real and Virtual Worlds*, Montpellier, France, 1996.

[Jot97] J.M. Jot. Efficient models for reverberation and distance rendering in computer music and virtual audio reality. In *International Computer Music Conference (ICMC 1997)*, Thessaloniki, Greece, 1997.

[JvSBC03] C. Jin, A. van Schaik, V. Best, and S. Carlile. Perceptual spatial-audio coding. In *2003 International Conference on Auditory Display*, Boston, MA, USA, 2003.

[JW95] J.M. Jot and O. Warusfel. Le spatialisateur. *GRAME*, 1995.

[Kac66] M. Kac. Can one hear the shape of a drum ? *American Mathematical Monthly*, 73:1–23, 1966.

[KBAA01] N. Korany, J. Blauert, and O. Abdel Alim. Acoustic simulation of rooms with boundaries of partially speacular reflectivity. *Applied Acoustics*, 62:875–887, 2001.

[Kel62] W.N Kellogg. Sonar system of the blind. *Science*, 137(3528):399–404, 1962.

[Ken94] G. S. Kendall. The effects of multi-channel signal decorrelation in audio reproduction. In *International Computer Music Conference (ICMC)*, 1994.

[Ken95] G. S. Kendall. The decorrelation of audio signals and its impact on spatial imagery. *Computer Music Journal*, 19(4):71–87, 1995.

[KJM03] B. Kapralos, M. Jenkin, and E. Milios. Auditory perception and spatial (3d) auditory systems. Technical Report Technical Report. CS-2003-07, Department of Computer Science, York University, Toronto, Ontario, Canada, 2003.

[KKFW99] A. Kaup, S. Khoury, A. Freed, and D. Wessel. Volumetric modeling of acoustic fields in cnmat's sound spatialization theatre. In *International Computer Music Conference (ICMC 99)*, 1999.

[KO83] K. Kurozumi and K. Ohgushi. The relationship between the cross-correlation coefficient of two-channel acoustic signals and sound image quality. *Journal of the Acoustical Society of America*, 74(6):1726–1733, 1983. wideness.

[KPT00] A.J. Kunkler-Peck and M.T. Turvey. Hearing shape. *Journal of Experimental Psychology: Human Perception and Performance*, 26(1):279–294, 2000.

[Kra94] G. Kramer. *Auditory Display, Sonification, Audification, and Auditory Interfaces*. Addison-Wesley, 1994.

[KTH99] C. Kyriakakis, P Tsakalides, and T. Holman. Surrounded by sound. *IEEE signal processing magazine*, (January 1999), 1999.

[Kut86] H. Kuttruff. *Room Acoustics*. Academic Press, 1986.

[KWC00] M. Kim, S. Wood, and L.-T. Cheok. Extensible mpeg-4 textual format (xmt), 2000.

[KWM93] G. S. Kendall, M. Wilde, and W. L. Martens. Apparatus and method for controlling the magnitude spectrum of acoustically combined signals, us patent 5,121,433, 1993.

[Kyr00] C. Kyriakakis. Virtual microphones and virtual loudspeakers for multichannel audio. *IEEE journal*, THPM 20.4, 2000.

[Lab03] A. Laborie. A new comprehensive approach of surround sound recording. In *114th AES Convention*, Amsterdam, The Netherlands, 2003.

[Lak] Lake. Dolby headphone product, http://www.lake.com.au/driver.asp?page=main/products/dolby

[Lak93] S. Lakatos. Recognition of complex auditory-spatial patterns. *Perception*, 22(3):363–374, 1993.

[Leh93] H. Lehnert. Auditoy spatial impression. In *AES 12th Conference*, pages 40–46, Copenhagen, Denmark, 1993.

[LI02] Y-W. Liu and J. O. Smith III. Perceptually similar orthogonal sounds and applications to multichannel acoustic echo cancelling. In *22nd AES conference*, Espoo, Finland, 2002.

[Lic48] J. C. R. Licklider. The influence of interaural phase relations upon the masking of speech by white noise. *Journal of the Acoustical Society of America*, 20(2):150–159, 1948.

[LL88] H. Lee and B.-H. Lee. An efficient algorithm for the image model technique. *Applied Acoustics*, 24:87–115, 1988.

[LML04] Listening to the mind listening, http://www.icad.org/websitev2.0/conferences/icad2004/concert.htm. 2004.

[LRYH97] R. Litovsky, B. Rakerd, T. Yin, and W. Hartmann. Psychophysical and physiological evidence for a precedence effect in the median sagitall plane. *Journal of Neurophysiology*, 77:2223–2226, 1997.

[Lun00] P. Lunden. Snd3d, a 3d sound system for vr and interactive applications. In *International Computer Music Conference (ICMC 2000)*, 2000.

[Mak62] Y. Makita. On the directional localisation of sound in the stereophonic sound field. *Eureopean Broadcasting Union journal*, 73:102–108, 1962.

[Mal92] D.G. Malham. Experience with large area 3-d ambisonic sound systems. *Proceedings of the Institute of Acoustics*, 14(5):209–215, 1992.

[Mal95] D.G. Malham. 3-d sound spatialization using ambisonic techniques. *Computer Music Journal*, 19(4):58–70, 1995.

[Mal99a] D. G. Malham. Spherical harmonic coding of sound objects - the ambisonic 'o' format. In *19th AES Conference*, Schloss Elmay, Germany, 1999.

[Mal99b] D.G. Malham. Higher order ambisonic systems for the spatialisation of sound. In *International Computer Music Conference (ICMC)*, 1999.

[Mas02] R. Mason. *Elicitation and measurement of auditory spatial attributes in reproduced sound*. Phd, University of Surrey, 2002.

[Max] Max. Max/msp, graphical programming language, www.cycling74.com.

[McG02] D. McGriffy. Visual virtual microphone, computer program, http://mcgriffy.com/audio/ambisonic/vvmic/, 2002.

[Men02] D. Menzies. W-panning and o-format, tools for object spatialization. In *ICAD 2002*, Kyoto, Japan, 2002.

[Mit03] Mitra. *Digital signal processing lab using matlab software*. McGraw-Hill, 2003.

[MM89] M. Morimoto and Z. Maekawa. Auditory spaciousness and envelopment. In *13th International Congress on Acoustics*, pages 215–218, 1989.

[Moo79] J. A. Moorer. About this reverberation business. *Computer Music Journal*, 3(2):13–28, 1979.

[Moo83] F.R. Moore. A general model for spatial processing of sounds. *Computer Music Journal*, 7(3):6–15, 1983.

[Mor02] M. Morimoto. The relationship between spatial impression and the precedence effect. In *International Conferenece on Auditory Displays (ICAD)*, Kyoto, Japan, 2002.

[MPE99] MPEG-4. Iso/iec 14496-1:1999 - coding of audio-visual objects - part 1: Systems. Technical report, ISO/IEC, 1999.

[MPE01] MPEG-4. Iso/iec 14496-1:2001 - coding of audio-visual objects - part 1: Systems. Technical report, ISO/IEC, 2001.

[MRK00] A. Mouchtaris, P. Reveliotis, and C. Kyriakakis. Inverse filter design for immersive auido rendering over loudspeakers. *IEEE transactions on Multimedia*, 2(2):77–87, 2000.

[MT02] C. Muller-Tomfelde. Hybrid sound reproduction in audio-augmented reality. In *AES 22nd International Conference on Virtual, Synthetic and Entertainment Audio*, 2002.

[MW02] J.D. Miller and E. M Wenzel. Recent developments in slab: a software-based system for interactive spatial sound synthesis. In *ICAD 2002*, Kyoto, Japan, 2002.

[Nae02] M. Naef. Spatialized audio rendering for immersive virtual environments. In *ACM VRST 2002*, 2002.

[NE98] R. Nicol and M. Emerit. 3d-sound reproduction over an extensive listening area: a hybrid method derived from holophony and ambisonic. In *16th Audio Engineering Conference*, 1998.

[Neu01] J. G. Neuhoff. Perceiving acoustic source orientation in three-dimensional space. In *ICAD 2001*, Espoo, Finland, 2001.

[Nie93] S.H. Nielsen. Auditory distance perception in different rooms. *Journal of the Audio Engineering Society*, 41(10):755–770, 1993.

[Nor05] M. J. Norusis. *SPSS 14.0 : advanced statistical procedures companion.* Prentice Hall, 2005.

[O'D03] M. O'Dwyer. *Development of a control and 3D display for 16 speaker 3D audio.* Masters, University of Wollongong, 2003.

[OFR02] L. Ottaviani, F. Fontana, and D. Rocchesso. Recognition of ditance cues from a virtual spatialization model. In *DAFX 2002*, Hamburg, Germany, 2002.

[Ope] OpenAL. Openal: Cross-platform 3d audio api, http://www.openal.org/.

[Orb70] R. Orban. A rational technique for synthesizing pseudo-stereo from monophonic sources. *Journal of the Audio Engineering Society*, 18(2):157–163, 1970.

[osc] Open sound control, cnmat website, www.cnmat.cnmat.berkeley.edu/osc/.

[OW97] A. Oppenheim and A. Willsky. *Signals and systems.* Prentice Hall signal processing series, 1997.

[PB82] D. R. Perrott and T. N. Buell. Judgments of sound volume: Effects of signal duration, level, and interaural characteristics on the perceived extensity of broadband noise. *Journal of the Acoustical Society of America*, 72(5):1413–1417, 1982.

[PB02a] G. Potard and I. Burnett. Proposal on sound source wideness and shape in mpeg audiobifs - m8533. In *Moving Picture Expert Group (MPEG) international conference*, Klagenfurt, Austria, 2002.

[PB02b] G. Potard and I. Burnett. Proposal on the use of digital item and digital item adaptation to transmit interactive 3d audio content - m8553. In *Moving Picture Expert Group (MPEG) international conference*, 2002.

[PB02c] G. Potard and I. Burnett. Refined descriptors for 3d audio dia - m8915. In *Moving Picture Expert Group (MPEG) international conference*, Shanghai, China, 2002.

[PB02d] G. Potard and I. Burnett. Using xml schemas to create and encode interactive 3-d audio scenes. In *DCW2002*, pages 193–202, Sydney, Australia, 2002.

[PB03] G. Potard and I. Burnett. A study on sound source shape and wideness. In *International Conference of Auditory Displays (ICAD)*, pages 25–28, Boston, USA, 2003.

[PB04a] G. Potard and I. Burnett. Control and measurement of apparent sound source width and its applications to sonification and virtual auditory displays. In *10th International conference on Auditory Displays (ICAD 2004)*, Sydney, Australia, 2004.

[PB04b] G. Potard and I. Burnett. Decorrelation techniques for the rendering of apparent source width in 3d audio displays. In *DAFX 2004*, Naples, Italy, 2004.

[PB04c] G. Potard and I. Burnett. An xml-based 3d audio scene metadata scheme. In *25th Audio Engineering Society Conference*, pages 102–112, London, UK, 2004.

[PBR95] J. M. Potter, F. A. Bilsen, and J. Raatgever. Frequency dependence of spaciousness. *Acta acustica*, 3:417–427, 1995.

[PE02] F. Pereira and T. Ebrahimi. *The MPEG-4 Book*. Prentice Hall PTR, 2002.

[Pel00] R.S. Pellegrini. Perception-based room-rendering for auditory scenes. In *Audio Engineering Society 109th Convention*, Los Angeles, USA, 2000.

[Pel01] R.S. Pellegrini. Quality assessment of auditory virtual environments. In *ICAD 2001*, Espoo, Finland, 2001.

[PI03] G. Potard and S. Ingham. Encoding 3d sound scenes and music in xml. In *International Computer Music Conference (ICMC2003)*, Singapore, 2003.

[PL03] K. Pihkala and T. Lokki. Extending smil with 3d audio. In *International Conference on Auditory Display (ICAD 2003)*, Boston, MA, USA, 2003.

[plu] Pluggo, cycling74, http://www.cycling74.com/products/pluggo.html.

[PMS80] D. R. Perrott, A. Musicant, and B. Scwethlem. The expanding-image effect: the concept of tonal volume revisited. *Journal of auditory research*, 20:43–55, 1980.

[PRB95] J. M. Potter, J. Raatgever, and F. A. Bilsen. Measures for spaciousness in room acoustics based on a binaural strategy. *Acta acustica*, 3:429–443, 1995.

[PS69] R. Plomp and H.J.M. Steeneken. Effect of phase on the timbre of complex tones. *Journal of the Acoustical Society of America*, 46:409–421, 1969.

[PS02] G. Potard and J. Spille. Report on mpeg-4 audiobifs sound source wideness core experiment - m8995. In *Moving Picture Expert Group (MPEG) international conference*, Shanghai, China, 2002.

[PS03] G. Potard and J. Spille. Study of sound source shape and wideness in virtual and real auditory displays. In *114th Audio Engineering Society (AES) Convention*, Amsterdam, 2003.

[PS04] G. Potard and G. Schiemer. Sonification of the coherence matrix and power spectrum of eeg signals. In *10th International conference on Auditory Displays (ICAD 2004)*, Sydney, Australia, 2004.

[PSS02] G. Potard, J. Seo, and J. Spille. Report of the second mpeg-4 audiobifs sound source wideness core experiment - m9171. In *Moving Picture Expert Group (MPEG) international conference*, Awaji Is., Japan, 2002.

[PSS03] G. Potard, J. Spille, and J. Seo. Report on sound source wideness 3rd core experiment - m9457. In *Moving Picture Expert Group (MPEG) international conference*, Pattaya, Thailand, 2003.

[Pul97] V. Pulkki. Virtual source positionning using vector base amplitude panning. *Journal of the Audio Engineering Society*, 45(6):456–466, 1997.

[Pul99] V. Pulkki. Uniform spreading of amplitude panned virtual sources. In *1999 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, 1999.

[Pul01] V. Pulkki. Spatial sound generation and perception by amplitude panning techniques, technical report 62. Technical Report 62, Helsinki Institute of Technology, 2001.

[Ray07] J.W.S. Rayleigh. Our perception of sound direction. *Philosophical Magazine*, 13:214–232, 1907.

[Ric16] G. J. Rich. A preliminary study of tonal volume. *Journal of experimental psychology*, 1:13–22, 1916.

[RK] K.W. Ross and J.F. Kurose. the tcp-friendly website, www-net.cs.umass.edu/kurose/transport/udp.html.

[Roc95] D. Rocchesso. The ball within the box: A sound-processing metaphor. *Computer Music Journal*, 19(4):47–57, 1995.

[Roc01] D. Rocchesso. Acoustic cues for 3-d shape information. In *ICAD 2001*, Espoo, Finland, 2001.

[RP76] R. M. Ruff and E. Perret. Auditory spatial pattern perception aided by visual choices. *Psychological Research*, 38:369–377, 1976.

[RS97] D. Rocchesso and J. O. Smith. Circulant and elliptic feedback delay networks for artificial reverberation. *IEEE Transactions on Speech and Audio Processing*, 5(1):51–63, 1997.

[Rum01] F. Rumsey. *Spatial audio*. Music technology series. Focal Press, Oxford, Boston, 2001.

[SC57] B. Sayers and E. C. Cherry. Mechanism of binaural fusion in the hearing of speech. *Journal of the Acoustical Society of America*, 29(9):973–987, 1957.

[Sch58] M.R. Schroeder. Artificial stereophonic effect obtained from a single audio signal. *Journal of the Audio Engineering Society*, 6(74), 1958.

[Sch70] M.R. Schroeder. Digital simulation of sound transmission in reverberant spaces. *Journal of the Acoustical Society of America*, 47(2), 1970.

[Sch02] M. Schreier. *Audio Server for Virtual Reality Applications*. Msc, Brunel University, 2002.

[Sea03] A. Sedes and et al. Egosound, an egocentric, interactive and real-time approach of sound space. In *DAFX 2003*, London, UK, 2003.

[sen] Gamecoda audio middleware, sensaura technology, http://www.sensaura.com/technology/index.php?article=middleware.htm.

[Sen99] J. J. Sendra. *Computational Acoustics in Architecture*. WIT Press, 1999.

[SFE00] J. Signes, Y. Fisher, and A. Eleftheriadis. Mpeg-4's binary format for scene description. *Signal Processing: Image Communication*, 15:321–345, 2000.

[SG04] G. Schiemer and Potard. G. Configurable hemisphere environment for spatialised sound. In *Australasian Computer Music Conference (ACMC 2004)*, Wellington, New Zealand, 2004.

[SH01] A. Sontacchi and R. Holdrich. Further investigations on 3d sound fields using distance coding. In *DAFX 2001*, Limerick, Ireland, 2001.

[SHLV97] L. Savioja, J. Huopaniemi, T. Lokki, and R. Vaananen. Virtual environment simulation - advances in the diva project. In *ICAD 97*, 1997.

[Sib02] A. Sibbald. Zoomfx for 3d-sound, sensaura white papers, www.sensaura.com. Technical report, 2002.

[SMI] SMIL. Synchronized multimedia.

[son] Sonic connections 2004, http://www.uow.edu.au/crearts/sonicconnections.html.

[SP82] J. Stautner and M. Puckette. Designing multi-channel reverberators. *Computer Music Journal*, 6(1):52–65, 1982.

[Spi03] J. Spille. Personal communication, 2003.

[spo] The audio spotlight, http://www.holosonics.com/.

[SSA02] J. Smith, S. Serafin, and J. Abel. Doppler simulation and the leslie. In *DFAX 2002*, Hamburg, Germany, 2002.

[Ste33] S. S. Stevens. *The volume and intensity of tones*. Phd thesis, Harvard university, 1933.

[Ste34] S. S. Stevens. The volume and intensity of tones. *American journal of psychology*, 46:397–408, 1934.

[Sun] Sun. Java3d api.

[SVH99] E.D. Scheirer, R. Vaananen, and J. Huopaniemi. Audiobifs: Describing audio scenes with the mpeg-4 multimedia standard. *IEEE Transactions on Multimedia*, 1(3):237–250, 1999.

[TB03] S. Tucker and G. J. Brown. Modelling the auditory perception of size, shape and material: Applications to the classification of transient sonar sounds. In *114th Audio Engineering Society Convention*, Amsterdam, The Netherlands, 2003.

[TDD+94] Craig C. Todd, Grant A. Davidson, Mark F. Davis, Louis D. Fielder, Brian D. Link, and Steve Vernon. Ac-3: Flexible perceptual coding for audio transmission and storage. In *96th Convention of the Audio Engineering Society*, page Preprint 3796, 1994.

[The80] G. Theile. *On the Localisation in the Superimposed Soundfield*. Phd, Technische Universitt Berlin, 1980.

[Tho52] G. J. Thomas. Volume and loudness of noise. *American journal of psychology*, 65:588–593, 1952.

[Tid01] D. Tidwell. *XSLT*. O'Reilly, 2001.

[Tit04] E. Tittel. *XML*. McGraw-Hill, 2004.

[TJ02] J.-M. Trivi and J.-M. Jot. Rendering mpeg-4 aabifs content through a low-level cross-platform 3d audioapi. In *IEEE International Conference onMultimedia and Expo*, pages pp. 513–516, Lausanne, Switzerland, 2002.

[TS01] A. Topol and F. Schaeffer. Enhancing sound description in vrml. In *International Computer Music Conference (ICMC 2001)*, La Havana, Cuba, 2001.

[TSA95] M. Tohyama, H. Suzuki, and Y. Ando. *The nature and technology of acoustic space*. Academic Press, London, 1995.

[Uni] Delft University. Wave field synthesis rendering system, http://www.soundcontrol.tudelft.nl/.

[Vaa03] R. Vaananen. User interaction and authoring of 3d sound scenes in the carrousso eu project. In *114th AES Convention*, page Preprint 5764, Amsterdam, The Netherlands, 2003.

[Vag01] H. Vagiione. Composing musical spaces by means of decorrelation of audio signals. In *DAFX 2001*, Limerick, Ireland, 2001.

[VB99] D. Vries and M.M. Boone. Wave field synthesis and analysis using array technology. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, New York, 1999.

[Vet04] A. Vetro. Mpeg-21 digital item adaptation: enabling universal multimedia access. *IEEE Multimedia magazine*, 11(1):84–87, Jan-Mar 2004 2004.

[VH99] R. Vaananen and J. Huopaniemi. Virtual acoustics rendering in mpeg-4 multimedia standard. In *International Computer Music Conference (ICMC) 1999*, 1999.

[VHP00] R. Vaananen, J. Huopaniemi, and V. Pulkki. Comparison of sound spatialization techniques in mpeg-4 scene description. In *International Computer Music Conference (ICMC) 2000*, 2000.

[Vor89] M. Vorlander. Simulation of the transient and steady-state sound propagation in rooms using a new combined ray-tracing/image-source algorithm. *Journal of the Acoustical Society of America*, 86(1):172–178, 1989.

[VRM98] VRML97. Iso/iec 14772-1. vrml97 standard. information technology - computer graphics and image processing - the virtual reality modeling language (vrml), part 1: Functional specification and utf-8 encoding. Technical report, ISO/IEC, 1998.

[VS] B. Vercoe and E. Scheirer. Saol: the mpeg-4 structured audio orchestra language. *Computer Music Journal*, 23(2):23–35.

[vst] Vst audio plugin technology, www.steinberg.net.

[VVHK97] R. Vaananen, V. Valimaki, J. Huopaniemi, and M. Karjalainen. Efficient and parametric reverberator for room acoustics modeling. In *International Computer Music Conference (ICMC)*, 1997.

[Wag90] W. M. Wagenaars. Localization of sound in a room with reflecting walls. *Journal of the Audio Engineering Society*, 38(3):99–110, 1990.

[WBS02] A. E. Walsh and M. Bourgues-Sevenier. *MPEG-4 Jump start*. Prentice Hall PTR, 2002.

[WDO97] J.-R. Wu, C.-D. Duh, and M. Ouhyoung. Head motion and latency compensation on localization of 3d sound in virtual reality. In *ACM VRST 97*, Lausanne, Switzerland, 1997.

[Wes98] J. R. West. *Five-Channel Panning Laws: An Analytical and Experimental Comparison.* Masters, University of Miami, 1998.

[win] Windome program, applied synergetics, http://www.applied-synergetics.com/ashp/html/windome_readme.html.

[WMK93] M.D. Wilde, W. L. Martens, and G. S. Kendall. Method and apparatus for creating de-correlated audio output signals and audio recordings made thereby, us patent 5,235,646, 1993.

[X3D] X3D. X3d: Open standards for real-time 3d communication - web3d consortium.

[Yan00] D. Yang. An inter-channel redundancy removal approach for high-quality multichannel audio compression. In *109th AES convention*, Los Angeles, USA, 2000.

[Yok85] Yokoyama. Device for forming a simulated stereophonic sound field, us patent 4,653,096, 1985.

[ZA02] U. Zolzer and X. Amatriain. *DAFX : digital audio effects.* Wiley, Chichester, England ; New York, 2002.

[Zah02] P. Zahorik. Auditory display of sound source distance. In *ICAD 2002*, Kyoto, 2002.

[Zic02] D. Zicarelli. How i learned to love a program that does nothing'. *Computer Music Journal*, 36(4):44–51, 2002.

Chapter 7

Appendix A

7.1 Measurements of inter-signal correlation

The cross-correlation coefficient is defined as the maximum absolute value of the cross-correlation function:

$$C_{xy} = max_{\tau}(|C_{x,y}(\tau)|) \tag{7.1}$$

Where $C_{x,y}(\tau)$ is the cross-correlation function between two signals $x(t)$ and $y(t)$ and is defined as:

$$C_{x,y}(\tau) = \lim_{T\to\infty} \frac{1}{2T} \int_{-T}^{+T} x(t)y(t+\tau)dt \tag{7.2}$$

The cross-correlation coefficient equals 1 for identical signals, 0 for totally decorrelated signals and -1 for identical signals but in phase opposition.

To measure the correlation coefficients between several signals it is convenient to place the correlation coefficients obtained by equations 7.2 and 7.1 in a coherence matrix defined as:

$$C_{i,j} = \begin{vmatrix} 1 & C_{1,2} & C_{1,3} & \dots & C_{1,k} \\ . & 1 & C_{2,3} & \dots & C_{2,k} \\ \vdots & \vdots & \ddots & \dots & \vdots \\ . & . & . & 1 & C_{k-1,k} \\ . & . & . & \dots & 1 \end{vmatrix} \tag{7.3}$$

This matrix is diagonal and symmetric. The aim of a decorrelation filterbank is thus to produce output signals that minimise the $C_{i,j}$ (non diagonal) correlation coefficients towards zero.

7.2 Matlab code for IIR decorrelation filter

```
%% IIR Decorrelation filter (all−pass)
%% Guillaume Potard 2003, University of Wollongong

function [wavout,polb,pola] = IIRDecor2(N,wav);  %N IIR filter order (must be even!!)

%A=rand(N/2,1)*0.9; % argument choose N number smaller than 0.9   %first half
%B=rand(N/2,1)*2*pi; % random phase

A=rand(N/2,1)*0.9; % argument choose N number smaller than 0.9   %first half
B=(rand(N/2,1)−0.5)*2*pi; % random phase

% make complex numbers
[real,imag]=pol2cart(B,A);

compli=real+j*imag; %

% second part is complex conjugates roots

compli((N/2)+1:N)=conj(compli);

% make denominator polynomial
```

```
pola=poly(compli);

% make numerator polynomial
% coefficients in reverse order to get all−pass response

polb=pola(length(pola):−1:1);

% filter input signal

wavout=filter(polb,pola,wav);
```

7.3 Matlab code for dynamic decorrelation filter

```
%%%%%

%% IIR dynamic Decorrelation filter based on lattice structure
%% Guillaume Potard 2003, University of Wollongong


function [wavout] = IIRdynamic6(wav,N,frame_len,sub_len);  %N IIR filter order (must
be even!!)

%%! Subframe must a mutlitple of frame length!

Zi=zeros(N,1);
A=rand(N/2,1)∗0.9; % argument choose N number smaller than 0.9   %first half
B=(rand(N/2,1)−0.5)∗2∗pi; % random phase

[real,imag]=pol2cart(B,A);
compli=real+j∗imag; %
compli((N/2)+1:N)=conj(compli);
compliold=compli; %last known good pole/zeros (no exiding circle unity)

index=1;
for i=1:(length(wav)/frame_len)−1% for each frame

 i;
```

```
deltaA=(rand(N/2,1)−0.5)∗0.9;
deltaB=(rand(N/2,1)−0.5)∗2∗pi; % random phase

A=A+deltaA;  %add small variations to magnitude and phase of poles
B=B+deltaB;


[real,imag]=pol2cart(B,A);
compli=real+j∗imag; %
compli((N/2)+1:N)=conj(compli);

for x=1:N
   if abs(compli(x))>0.95     % if norm exedes 0.95, revert to good complex vector
      compli(x)=compliold(x);
   end
end

compliold=compli; %save good vector

pola=poly(compli); polb=pola(length(pola):−1:1);
%zplane(1,pola);                    %Plot pole zeros
%pause(0.1)
% filter input signal

%wavout(index:index+frame_len)=filter(polb,pola,wav(index:index+frame_len));
%[wavout(index:index+frame_len),Zf]=filter(polb,pola,wav(index:index+frame_len),Zi);

[k,v]=tf2latc(polb,pola);%compute lattice ceoffs from FIR/IIR coeffs

[wavout(index:index+frame_len),g,Zf]=latcfilt(k,v,wav(index:index+frame_len),'ic',Zi);
%%WITH HISTORY

%[wavout(index:index+frame_len),g]=latcfilt(k,v,wav(index:index+frame_len));

Zi=Zf; index=index+frame_len; end
```

Chapter 8

Appendix B

8.1 List of DSP layer commands

8.1.1 Sound source control

source <source ID> <command> <..> <..> <..> ..... :

<source ID> : a number from 0−9999 corresponding to the number used in creating the source.

<command> : `'pos_cart'` <x> <y> <z>

Used to move the position of the source in cartesian coordinates. The <x>, <y>, <z> coordinates are specified as floating point variables.

`'pos_sph'` <azimuth> <elevation> <distance>

Used to move the position of the source in sperical coordinates. The <azimuth> and <elevation> are float

variables spcified in degrees. Distance is also a float variable.

`'play'` <0 or 1>

Used to play or stop the sound associated with a source.

0 to stop, 1 to play.

`'loop'` <0 or 1>

Used to enable or disable looping of a sound. 0 to disable

loop, 1 to enable loop.

`'gain'` <value>

Adjusts the gain of a source. Is a float value ranging from 0 to 5.

8.1.2 Reflective surface control

Surface <command> <surface no> <..> <..> ...

<command> : 'Coords' <surface no> <x1> <y1> <z1> <x2> <y2> <z2> <x3> <y3> <z3> <x4> <y4> <z1> Sets the vertex coordinates for the surface to the specified coordinates all of which are float variables.

8.1.3 Room reverberation control

room <command> <..> <..> <..> ..... :

<command> : `'room_pres'` <value>

`'run_reverb'` <value>

`'envelop'` <value>

`'heavy'` <value>

`'liveness'` <value>