



## Full Reviewed Paper at ICSA 2019

Presented\* by VDT.

### The Distribution of Ambisonic and Point Source Rendering to Ethernet AVB Speakers

S. Devonport<sup>1</sup>

R. Foss<sup>2</sup>

<sup>1</sup> Rhodes University, South Africa, Email: [tonetechnician@gmail.com](mailto:tonetechnician@gmail.com)

<sup>2</sup> Rhodes University, South Africa, Email: [r.foss@ru.ac.za](mailto:r.foss@ru.ac.za)

#### Abstract

Point source rendering is used by many object-based audio systems to mix audio objects to loudspeaker arrangements. Algorithms such as Distance-Based Amplitude Panning and Vector-Base Amplitude Panning allow for audio objects to have their locations rendered with high precision. It has been shown that in the context of loudspeaker rendering, point sources rendered with Ambisonics are often spatially blurred. However, Ambisonics does have the advantage of being able to create interesting spatial audio effects and ambient scenes can be recorded using Ambisonic microphones. This paper intends to highlight the advantages that may be gained by combining Ambisonics with virtual point source rendering. It is well known that the processing required for rendering both point source and Ambisonics can have a large overhead. To mitigate this, a distributed spatial audio system based on Ethernet AVB and distributed endpoint processors is modified to incorporate both point source rendering and Ambisonics. An example is given of how point source rendering can be integrated with Ambisonics using this system with existing software.

#### 1. Introduction

3D immersive audio can be described as the process of creating and rendering spatial audio content to a loudspeaker arrangement or headphones. With the advent of consumer VR/AR systems, there is a need for new tools that are able to render both accurate audio objects and spatially realistic ambience. Ambisonics has become a defacto standard for VR/AR productions and content is now being widely released in Ambisonic format designed mostly for headphone listening. When using Ambisonics in headphone listening, point sources are able to be rendered with high precision however there are some aspects of headphone listening that may cause breaks in immersion due to the fact that:

1. Headphones by design are unable to reproduce subsonic frequencies that enhance immersion.
2. Headphone based listening is exclusive to the person wearing the headphones.

One problem that could prevent Ambisonics from being more accepted in consumer multichannel speaker systems is the spatial precision lost due to spatial blur induced by the limitation of the loudspeaker configuration and the listening environment [1]. As such, when rendering precise virtual point sources to loudspeakers, it would be preferable to use a virtual point source rendering algorithm such as distance-based amplitude panning (DBAP) or vector-base amplitude panning (VBAP) [2] [3]. Whilst Ambisonics can cause spatial blur of encoded virtual point sources, this can be less problematic for Ambisonic recording and spatial ambience effects. It seems appropriate then to use point source panning in tandem with Ambisonics audio.

Point source panning using VBAP allows track objects to be localized precisely in loudspeaker layouts using loudspeaker triplets. These track objects can be fed signals that would normally be fed to actual speakers and these objects can now be considered 'virtual loudspeakers'. This concept has already

been used in the All Round Ambisonic Decoding (AllRAD) technique which uses VBAP generated point sources to generate an ideal loudspeaker layout for Ambisonics decoding, however, it can also be applied to surround sound speaker layout remapping [4]. This allows standard surround sound content to be played back alongside Ambisonics and object-based audio, on irregular speaker environments. Not only does this create interesting creative possibilities, but it also allows for backwards compatibility with channel-based surround sound content that is currently available in many movies and games.

The ImmerGo spatial audio workstation provides a framework to implement such features [5]. It already incorporates object-based point source panning with DBAP and VBAP, and can be modified to incorporate Ambisonic rendering as it uses object metadata to describe loudspeaker positions. Its client-server based distributed architecture decouples its usage from particular sound source software such as a digital audio workstation (DAW), and this enables it to be easily integrated into any audio project's workflow. The use of Ethernet AVB and distributed network processors mitigates the processing demands of these different rendering algorithms and makes it scalable to any number of loudspeakers [6].

The research in this paper modifies ImmerGo to include both distributed virtual point source rendering and Ambisonics rendering. The VBAP algorithm is utilized to provide speaker remapping that allows for playback of surround sound content on irregular loudspeaker layouts and the AllRAD algorithm is used to decode up to 4<sup>th</sup> order Ambisonics to irregularly spaced loudspeakers.

This paper will continue with an overview of channel-based audio, object-based audio and Ambisonics, highlighting key benefits of each. This will be followed by an explanation of how these immersive approaches were merged within ImmerGo to create a system incorporating their combined benefits. Finally, there will be a description of an installation that utilizes this system.

## 2. Representations of 3D Immersive Audio

Channel-based audio (CBA), object-based audio (OBA) and scene-based audio (SBA) are representations used in current state-of-the-art immersive audio rendering systems. These have been developed to simplify the process of creating and distributing spatial audio content correctly to different loudspeaker arrangements. Each of these representations lends themselves to different rendering procedures that are required to generate and feed audio sources to loudspeakers. Currently, there are a few systems available that have been developed to render these different representations alongside each other, notably the MPEG-H Renderer, the European Broadcast Union (EBU) ADM Renderer (EAR) which is now being implemented in the ITU-R BS.2127 [7] [8] [9] [10].

### 2.1. Channel-Based Audio

CBA can be considered a 'loudspeaker first' spatial audio technique. A content creator will mix composition into a multichannel wav file. It assumes the loudspeaker arrangement used for playback will be the same loudspeaker arrangement that was used when generating the mix. This has led to the well-known ITU surround sound loudspeaker arrangements used in home theatre [11]. This format requires a relatively simple rendering procedure, as there is a one to one mapping between audio channel and loudspeaker.

While the CBA format has worked relatively well in consumer audio distribution, it has avoided dealing with the problem of non-standard loudspeaker configurations: if you play a CBA mix to an irregular layout of loudspeakers, there is a good chance it will not sound the way the content creator intended. There are solutions available that mitigate this such as the MPEG-H renderer which provides tools that can translate between different CBA layouts [12].

### 2.2. Object-Based Audio

Object-oriented distribution formats have been developed to allow for spatial audio playback to be compatible with any loudspeaker arrangement by using powerful processors that render the spatial audio content at playback time. The object-based format incorporates audio sources, the positional information of these sources and the loudspeaker playback environment. This has been aptly named OBA. OBA represents each audio channel associated with location, spread and directionality metadata. When creating object-based audio content, the metadata allows the spatial scene to be rendered correctly on non-standard and standard loudspeaker arrangements alike [13]. At the playback stage, the metadata associated with an audio channel is fed into an algorithm that generates the loudspeaker feeds so that the audio channel is correctly positioned in 3D space.

OBA has been incorporated in systems such as Dolby Atmos, DTS:X and Auro3D. It comprises a significant part of the MPEG-H 3D Audio standard. The EBU has released an open source Audio Definition Model (ADM) metadata format that has been a resource for the ITU Audio Definition Model [14] [15] and is used by the open source EAR renderer and the ITU-R BS.1770 [16].

There are a variety of object-based rendering algorithms that can be used to render audio object positions according to their metadata, most notably VBAP and DBAP. While both these algorithms render audio objects using metadata there are differences in how they render the audio in relation to the listening position. VBAP assumes there to be a listener centred at an origin point or sweet spot, whilst DBAP does not assume this.

### 2.3. Scene-Based Audio and Ambisonics

SBA could be considered a combination of CBA and OBA [9]. It encodes an infinite number of audio objects into a known set of audio channels known as the *audio scene*. This encoded format is loudspeaker agnostic and the format must be decoded at the loudspeaker endpoint to be heard correctly

[17]. There are different formats that describe the Ambisonic soundfield, however, it has become a standard to use the AmbiX format which is output by various plugins and is the format used in the research in this paper [18] [19].

Ambisonics uses a set of encoding functions based on the spherical harmonic transform functions [20]. These functions are used to encode the positions of audio objects that lie on a sphere into a finite set of audio channels called the Ambisonic soundfield. An Ambisonic soundfield can be captured using Ambisonic microphones, or synthesized by multiplying a mono channel by each encoder function to form the Ambisonic encoded audio channels.

The number of channels in the encoded signal is proportional to the order according to:

$$\text{channels} = (\text{order} + 1)^2$$

1<sup>st</sup> order content is sometimes referred to as B-format and has four channels, and higher order content has more, with higher order content having a higher spatial resolution [21].

An in-depth discussion into the decoder formulation is omitted as it has already been covered in depth in numerous other publications [22]. However, it is important to know that at the decoding stage, these audio channels are summed together and fed to loudspeakers according to decoder scaling values that are calculated according to the loudspeaker position. These decoder values ensure that all the loudspeakers are playing audio from the Ambisonic signal, with the sound scene directionality being induced by the weighting and phase inversions of each audio signal from the encoded Ambisonic soundfield. Since all the loudspeakers are playing at once, the decoding can cause some spatial blur when listening to Ambisonic content on loudspeaker arrays with insufficient, incorrectly spaced loudspeakers, or when listening at an off-centre position [23].

Due to the geometrical nature of Ambisonics, the encoded channels are also able to be manipulated and transformed efficiently in real-time using processing matrices. This is particularly useful in headtracked environments for headphone reproduction of VR audio. This feature has also allowed for interesting and efficient Ambisonic effects to be created [24]. These effects are easily incorporated into workflows currently used for media production [25]. As well as this, there are a variety of Ambisonic based processors that have been developed for both game audio engines<sup>1</sup> and digital audio workstations<sup>2,3</sup>.

In the last few years, many Ambisonic microphone arrays have been released that can be used to record spatial audio that can be played alongside VR games and video. The majority of these microphones are in a tetrahedral arrangement that records in A-format, which is converted to the B-format used

in Ambisonics. These microphones capture ambience and directionality sufficiently accurately, however higher order microphones capture sound field directionality more accurately. There is also the well-known Eigenmike<sup>4</sup> which is a 32-channel microphone array that is able to generate higher order Ambisonic files up to 4<sup>th</sup> order. Some free-to-download Ambisonic recordings made with the Eigenmike known as the Eigenscape can be found online [26].

Of considerable interest is that these microphones are able to capture a 3D directional impulse response of an environment that can be used as a convolution filter for other Ambisonic encoded content. These so-called Directional Room Impulse Responses (DRIRs) provide accurate 3D modelling of acoustic spaces [27]. Recently, a few databases of DRIRs have been converted into the Spatial Oriented Format for Acoustics (SOFA) convention [28] [29]. SOFA provides a standardized format which can be used interchangeably between different systems. This provides a promising basis for the growth of these applications in the future.

### 3. ImmerGo Spatial Audio Workstation

The ImmerGo spatial audio workstation provides a client-server-based approach to immersive audio rendering and is built on web technologies [5]. It allows a user to render the location of multiple virtual point sources in an environment using any device with a browser. ImmerGo's approach to immersive audio rendering employs a distributed processing model that moves the final audio rendering processing out to multiple endpoint processors attached to loudspeakers. Each processor is dedicated to the loudspeaker it is attached to. This ensures a scalable solution, as any loudspeaker added to the loudspeaker array also incorporates the additional processing power.

As shown in **Fig. 1**, a user is able to select a track and control its position, level and spread angle within the loudspeaker array. ImmerGo also can control a DAW transport using an internal MIDI bus. As well as this, track object metadata parameters are able to be recorded and automated according to the internal clock or MIDI timecode from an external source. When playing this automation, the ImmerGo track object model is updated according to MIDI time code quarter frames at roughly 8ms intervals.

#### 3.1. ImmerGo Track and Room Object Model

Shown in **Table 1** is ImmerGo's track object model. This model contains parameters which are used by the renderer to generate mixing values used for the endpoint processors that mix the spatial composition correctly. The dynamic parameters are able to be changed in real-time using the ImmerGo UI. The track ID does not change.

<sup>1</sup>  
[https://www.audiokinetic.com/library/edge/?source=Help&id=using\\_ambisonics\\_in\\_plugins](https://www.audiokinetic.com/library/edge/?source=Help&id=using_ambisonics_in_plugins)

<sup>2</sup> IEM plugins - <https://plugins.iem.at/>

<sup>2</sup> SPARTA/COMPASS plugins - [http://research.spa.aalto.fi/projects/sparta\\_vsts/plugins.html](http://research.spa.aalto.fi/projects/sparta_vsts/plugins.html)

<sup>4</sup> Eigenmike - <https://mhacoustics.com/products>

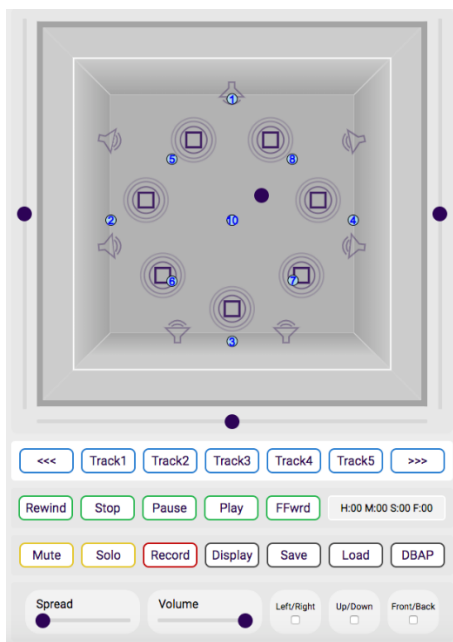


Fig. 1: The ImmerGo Client.

Track Object Model	
- Position (x,y,z)	Dynamic
- Spread	Dynamic
- Volume	Dynamic
- Track ID	Static
- Mute	Dynamic
- Solo	Dynamic
- Selected	Dynamic

Table 1: The ImmerGo Track Object Model

The loudspeaker object model shown in Table 2 is used by the spatial audio renderer to calculate the correct loudspeaker signals for a particular audio channel. Loudspeakers are able to be positioned using the ImmerGo UI which generates the loudspeaker object metadata. Each audio channel fed to the loudspeaker is able to be scaled and delayed. In this research, only the scaling function is used.

Loudspeaker Model	
- Position (x,y,z)	
- Mix values for each audio channel	
- Delay levels for each audio channel	

Table 2: The ImmerGo Loudspeaker Model

### 3.2. Ethernet AVB

The loudspeaker processors interface to an Ethernet AVB network. The Ethernet AVB standard provides the framework to allow for a synchronous real-time audio network. It adds two standards on top of three older standards to enable the appropriate quality of service for real-time audio delivery and control. These are the 1722 Audio Video Transport protocol (AVTP) and the 1722.1 Audio Video Discovery, Enumeration, Control and Connection Management protocol (AVDECC) [30] [31].

These provide two important benefits that are used extensively by the ImmerGo system:

1. AVTP provides the ability to stream multichannel audio synchronously to multiple loudspeaker processors distributed on a network.
2. AVDECC provides the ability to control multiple loudspeaker processors distributed on a network in real-time using the AVDECC Enumeration and Control Protocol (AECp).

### 3.3. Distributed Endpoint Processors

Each distributed loudspeaker processor used by ImmerGo contains an XMOS microcontroller and SHARC DSP chip [32] [33]. This provides each loudspeaker processor with the capability to have audio mixed in real-time according to the output of various spatial audio rendering algorithms housed within the ImmerGo server. Furthermore, the processors and speakers are able to be powered over Ethernet using the PoE+ protocol.

A core component of the endpoint processor is its multi-in multi-out (MIMO) mixer matrix that is controlled using the AECp protocol. The MIMO mixer can scale and phase invert each of the 32 channels of audio coming from the Ethernet AVB stream before summing them to either two loudspeakers attached to the processor. The values used to scale the output are generated by the ImmerGo server according to object metadata.

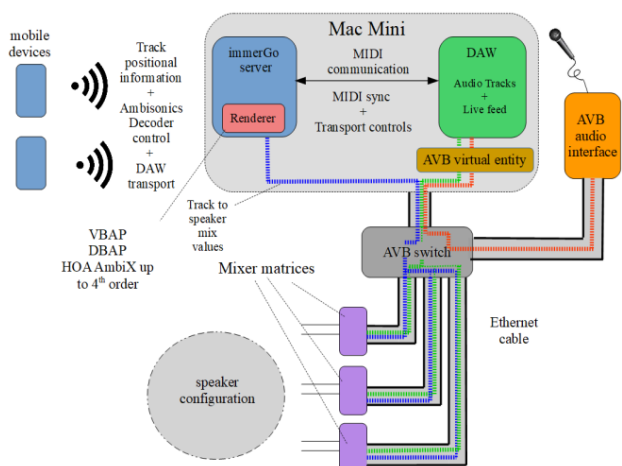
### 4. Modifications to ImmerGo

As shown in Fig. 2 below, ImmerGo’s UI and spatial audio renderer is modified to include the capability to control and render higher order Ambisonics alongside VBAP and DBAP. This allows for a variety of Ambisonics recordings and effect chains to be played alongside virtual point source rendered content. Other AVB interfaces with live feeds are also able to be included in the network using the native Apple AVB virtual entity.

We see in orange the live audio feed from a microphone passing through the network to the DAW. Within the DAW, the live feed can be processed and streamed out alongside other audio content housed within the DAW shown in green. When a user interacts with a track object, Ambisonics decoder or speaker remapping function on the UI, the updates are sent over a web socket which is then parsed within the server. The server’s renderer then uses these parameters to render mixer values for the loudspeaker processor mixer matrices which are sent to the endpoint processors using AVDECC AECp messages.

There were three main considerations taken into account when implementing these modifications:

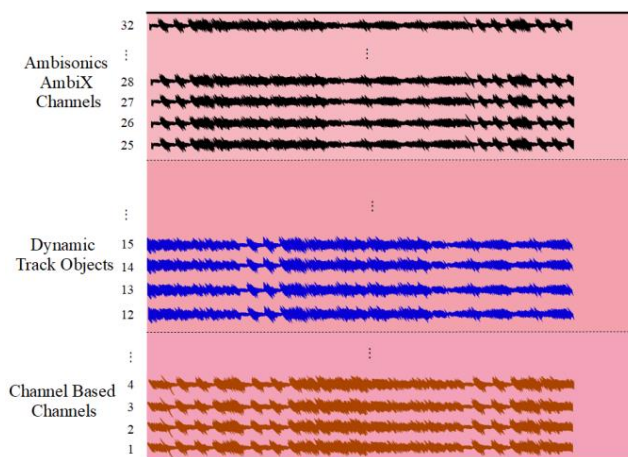
1. The mixer matrix limit of 32 channels.
2. Changes to listening position across different speaker environments.
3. Speaker remapping using point source rendering.



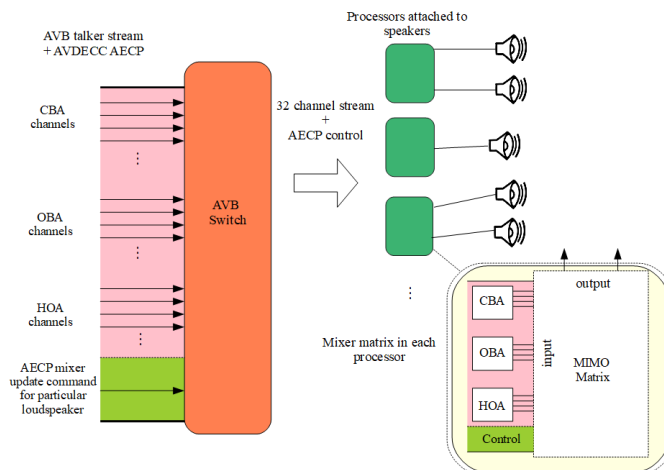
**Fig. 2:** Modifications to the ImmerGo system to include Ambisonics decoding and speaker remapping control.

### 4.1. Audio Transport Allocation to the Mixer Inputs

As shown in Fig. 3, the audio transport bus is split into three sub-busses that are fed the appropriate audio channels pertaining to the style of rendering being performed. The audio channel allocation is changed dynamically according to the number of channels required for each rendering algorithm. Because of this, there is a channel allocation trade-off. The higher the order of Ambisonic decoding, the less available channels there are for the point source rendering that is used for OBA and CBA content. This is realised in how a user can interact with the UI. If a part of the audio transport is dedicated to Ambisonics, the user is unable to interact with that channel using the typical track object controls. However, if a part of the audio transport is dedicated to channel-based content, the user is able to fine tune the virtual speaker object position.



**Fig. 3:** Channel allocation from the audio source to the Ethernet AVB stream.



**Fig. 4:** Channel allocation of AVB stream with control message from ImmerGo to each endpoint mixer matrix.

Fig. 4 above shows how each endpoint mixer matrix input is fed according to the channel allocation given by the sub-bus components for each rendering algorithm. The channel-based audio that renders surround sound content was placed first, with object-based speaker remapping applied. Then dynamic object-based tracks were placed second and the rest of the transport was dedicated to the Ambisonic encoded signals. The mixer inputs dedicated to each format then had their mixer crosspoints updated according to the correct values pertaining to the rendering algorithm.

### 4.2. Surround Sound Speaker Remapping

As has been shown by the AllRAD approach, virtual point sources created using VBAP are able to be used as 'virtual loudspeakers' [4]. This concept is used to provide ImmerGo with the capability to playback channel-based content. By assigning a particular loudspeaker signal from the CBA content to a track object, the loudspeaker feed is able to be played from that position. In order to achieve this in practice, a central origin is needed so that each object is rendered to the correct location in the array. This origin position is known as the listening position and is calculated as the midpoint of the maximum and minimum (x,y,z) locations of the loudspeakers in the array.

DBAP would also be able to pan a virtual source, however, the algorithm is based on loudspeaker energy distribution such that all the loudspeakers are required to play the audio to keep a constant energy level. VBAP provides more precise point sources when compared to DBAP since only 3 speakers are active at once.

The option to 'remap speakers' is provided in the user interface as shown in Fig. 5. When selecting this option, controls are shown that are used to set the desired loudspeaker configuration using virtual point sources panned with VBAP, as well as control the bass management level. A selection of mono, stereo, 2.1, 4.1, 5.1 and 7.1 are currently available. The vertical offset of the listening position can be controlled using the vertical offset slider. This is used in case the virtual source positions are not in the same plane as the listener position. The



track buttons are then changed to give the user feedback about which surround channels are mapped to which location.

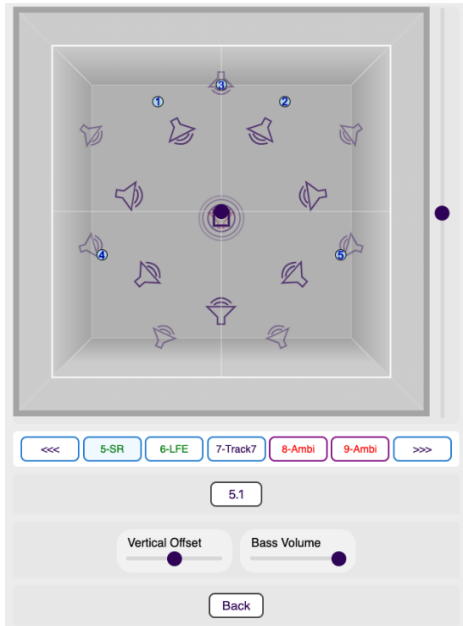


Fig. 5: Speaker remap control using ImmerGo.

### 4.3. Ambisonics Decoder Control

Traditional Ambisonics requires very strict loudspeaker layouts such as cubes or icosahedrons however in recent years, the AllRAD approach has been developed to overcome this limitation. It makes use of virtual point source panning using VBAP to create ‘virtual loudspeakers’ in the layout of T-designs that Ambisonics encoded signals are able to map to successfully [4]. This solution allows for Ambisonics to be decoded to irregular loudspeaker layouts. In order to have ImmerGo compatible with different layouts, this approach was employed.

ImmerGo was modified to incorporate the Spatial Audio Framework (SAF) <sup>5</sup> which has recently been developed at Aalto University and provides an AllRAD solution. SAF is able to generate AllRAD Ambisonics decoder values with maxRE weightings up to 7<sup>th</sup> order AmbiX for any speaker layout. On suboptimal layouts where the convex hull calculation results in large changes of gain values, an ‘imaginary’ loudspeaker is added either directly above or below the loudspeaker array to ensure a balanced energy distribution [4].

In order to use the features provided in SAF, NodeJS bindings were developed that allowed for data to be passed between the ImmerGo server’s NodeJS runtime and the SAF library. Due to the limit imposed by the endpoint mixer matrices, a maximum of 4<sup>th</sup> order (25 channels) was allowed.

When solving for the loudspeaker directions the listening position is selected as the centre of the loudspeaker array. From there, vectors are drawn to the loudspeakers and these

are used to solve for the loudspeaker directions. Each loudspeaker direction is then passed to SAF which calculates the AllRAD decoder values. These values are then sent to the endpoint mixer matrices at which point the track object control in the ImmerGo UI becomes disabled to the user. The Ambisonic decoder controls are found in a sub menu.

ImmerGo’s Ambisonic controls are shown in Fig 6. below. Loudspeaker environments may change and as such, the listening position also needs to be adjustable. If the initial point of origin is incorrect for the loudspeaker array, the sound scene can sound weighted unevenly in the vertical plane. As such, the ability to change the listening position vertical offset is provided, and is different from the vertical offset parameter of the speaker remapping option. When changing this control, the AllRAD decoders are recalculated, with the loudspeaker directions shifted according to the change in the origin. The result is the impression that the origin of the Ambisonic soundfield has shifted vertically in the loudspeaker array. Along with these controls is the option to convert from 3D normalised and a semi-normalized (N3D or SN3D) normalization scheme [4].

The Ambisonic soundfield is also able to be rotated using the relevant yaw, pitch and roll sliders. A bass management control is provided allows for changes to the volume of the omnidirectional component of the Ambisonic encoded signal feeding the subwoofers.

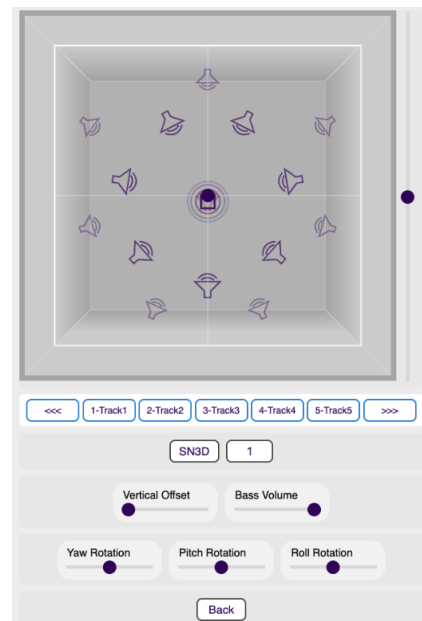


Fig. 6: Ambisonics decoder control using the ImmerGo UI.

## 5. Combining Ambisonics with Virtual Point source Rendering

To demonstrate the benefits of combining these algorithms, a system that combines the modified ImmerGo system with the Reaper DAW has been created, although any other audio software could be used. A score was built that made use of

<sup>5</sup> [https://github.com/leomccormack/Spatial\\_Audio\\_Framework](https://github.com/leomccormack/Spatial_Audio_Framework)

point source panning, speaker remapping and Ambisonics. It consisted of a male voice, Ambisonic nature recording, 5.1 sound effects, and track objects that were used to move particular sound sources within the Ambisonic scene.

The score intended to recreate more precisely the sound of a voice being inside a natural soundscape. Ambisonic recordings of a nature scene and DRIRs were captured, using the H3-VR microphone. This recording was used alongside the point source and 5.1 content. Mixed into the Ambisonic bus were other audio effects generated by Ambisonic plugins to create spatial echoes and virtual source position modulations, some of which were convolved with Ambisonic DRIRs taken in different environments.

The channel allocation for the content in this score is:

1. **Channels 1-6:** Previously created 5.1 sound effects including close miked natural sounds such as leaves rustling and animal sounds.
2. **Channels 7-16:** Track with different spot mikes of birds and bug sounds.
3. **Channels 29-32:** 1<sup>st</sup> order AmbiX recordings from the H3-VR microphone mixed with other 1<sup>st</sup> order Ambisonic spatial effects of birds.

The project was run as follows:

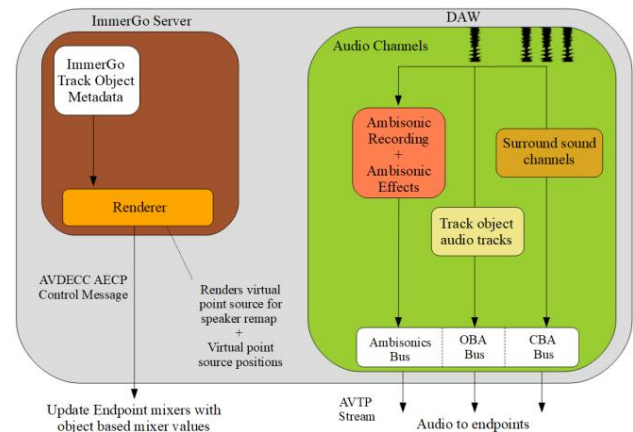
1. At startup, the speaker configuration was input to the ImmerGo system using the mobile device GUI.
2. These positions were used by the SAF to calculate and update the endpoint mixers for the Ambisonic decoders according to the loudspeaker positions
3. The speaker remapping was set to use 5.1 rendering.
4. The Ambisonic decoder was adjusted according to the content's format.
5. The audio content was played from the DAW and track objects were able to be controlled alongside the Ambisonic and surround sound content.

This system highlights some key benefits when combining these various techniques to create an immersive audio soundscape. The following capabilities are provided:

- Virtual point source panning using DBAP and VBAP.
- 5.1 surround sound content playback.
- 3D Ambisonic recordings for the rendering of spatial audio recordings.
- Ambisonics effect processing for efficient spatial audio effects.

**Fig. 7** below shows how these were combined using ImmerGo along with the Reaper DAW. The Ambisonic decoder was able to decode the Ambisonics recordings and effects that were summed into the Ambisonics transport bus. As well as this, the speaker remapping option allowed for the channel-based content to be rendered on the irregular layout.

Furthermore, individual track objects were able to be moved around the speaker array according to automation tracks.



**Fig. 7:** Object metadata controlling the ImmerGo point source renderer alongside Ambisonics rendering.

## 6. Conclusion

This paper has covered the various representations of immersive audio with their associated rendering algorithms. This information was used to modify the ImmerGo spatial audio workstation so that it is able to perform rendering for channel-based, object-based and scene-based audio alongside each other. In particular, sub menus were added that provide the necessary controls for speaker remapping and Ambisonic decoder setup. A possible project layout that incorporates the rendering of channel-based, object-based and Ambisonic content alongside one another is given. This project highlights how a rich combination of channel-based audio, point source panning and Ambisonics effects can be rendered simultaneously.

## 7. References

- [1] G. Marentakis, F. Zotter and M. Frank, "Vector-Base and Ambisonic Amplitude Panning: A Comparison Using Pop, Classical, and Contemporary Spatial Music," *Acta Acustica united with Acustica*, vol. 100, no. 5, pp. 945-955, 2014.
- [2] T. Lossius, P. Baltazar and T. Hogue, DBAP--distance-based amplitude panning, Ann Arbor, MI: Michigan Publishing, University of Michigan Library, 2009.
- [3] V. Pulkki, "Virtual Sound Source Positioning Using Vector Base Amplitude Panning," *J. Audio Eng. Soc.*, vol. 45, pp. 456-466, 1997.
- [4] F. Zotter and M. Frank, "All-round ambisonic panning and decoding," *Journal of the Audio Engineering Society*, vol. 60, pp. 807-820, 2012.
- [5] R. Foss and A. Rouget, "Immersive Audio Content Creation Using Mobile Devices and Ethernet AVB," in *Audio Engineering Society Convention 139*, 2015.
- [6] S. Devonport and R. Foss, "An Investigation into the Distribution of 3D Immersive Audio Renderer Processing to Speaker Endpoint Processors," in *VDT Tonmeistertagung*, Cologne, 2018.

- [7] A. Murtaza, J. Herre, J. Paulus, L. Terentiv, H. Fuchs and S. Disch, "ISO/MPEG-H 3D Audio: SAOC 3D Decoding and Rendering," in *Audio Engineering Society 139*, New York, 2015.
- [8] European Broadcast Union, "Tech 3388: ADM Renderer for Use in Next Generation Audio Broadcasting," EBU, Geneva, 2018.
- [9] S. Shivappa, M. Morrell, D. Sen, N. Peters and S. M. A. Salehin, "Efficient, Compelling, and Immersive VR Audio Experience Using Scene Based Audio/Higher Order Ambisonics," in *AES International Conference on Audio for Virtual and Augmented Reality*, Los Angeles, 2016.
- [10] International Telecommunications Union, "ITU BS.2127: Audio Definition Model renderer for advanced sound systems," 2019.
- [11] International Telecommunications Union (ITU), "ITU-Report BS.2159-6," International Telecommunications Union (ITU), Geneva, 2015.
- [12] J. Herre, J. Hilpert, A. Kuntz and J. Plogsties, "MPEG-H 3D Audio—The New Standard for Coding of Immersive Spatial Audio," *IEEE Journal of Selected Topics in Signal Processing*, vol. 9, no. 5, pp. 770-779, 2015.
- [13] R. Bleidt, A. Borsum, H. Fuchs and S. M. Weiss, "Object-Based Audio: Opportunities for Improved Listening Experience and Increased Listener Involvement," 2014.
- [14] European Broadcasting Union, "TECH 3364 Audio Definition Model Metadata v2.0," European Broadcasting Union, Geneva, 2018.
- [15] International Telecommunications Union, "Recommendation ITU-R BS.2076-1," International Telecommunications Union, Geneva, 2017.
- [16] International Telecommunications Union, "ITU-R BS.1770: Algorithms to measure audio programme loudness and true-peak audio level," 2015.
- [17] N. Peters, D. Sen, M.-Y. Kim, O. Wuebbolt and S. M. Weiss, "Scene-based Audio Implemented with Higher Order Ambisonics," *SMPTE Motion Imaging Journal*, vol. 125, no. 9, pp. 16 - 24, 2016.
- [18] C. Nachbar, F. Zotter, E. Deleflie and A. Sontacchi, "AmbiX - A Suggested Ambisonics Format," Lexington, 2011.
- [19] L. McCormack and A. Politis, "SPARTA & COMPASS: Real-Time Implementations of Linear and Parametric Spatial Audio Reproduction and Processing Methods," in *AES International Conference on Immersive and Interactive Audio*, York, 2019.
- [20] M. A. Gerzon, "Periphony: With-height Sound Reproduction," *Journal of the Audio Engineering Society*, vol. 21, no. 1, pp. 2-10, 1973.
- [21] S. Bertet, J. Daniel, L. Gros, E. Parizet and O. Warusfel, "Investigation of the Perceived Spatial Resolution of Higher Order Ambisonics Sound Fields: A Subjective Evaluation Involving Virtual and Real 3D Microphones," in *AES 30th International Conference: Intelligent Audio Environments*, Finland, 2007.
- [22] D. Artega, "Lecture Notes: Introduction to Ambisonics," in *Audio 3D – Grau en Enginyeria de Sistemes Audiovisuals*, Universitat Pompeu Fabra, 2015.
- [23] L. S. R. Simon, H. Wuethrich and N. Dillier, "Comparison of Higher-Order Ambisonics, Vector- and Distance-Based Amplitude Panning using a hearing device beamformer," in *4th International Conference on Spatial Audio*, Graz, 2017.
- [24] D. Rudrich, F. Zotter and M. Frank, "Efficient Spatial Ambisonic Effects for Live Audio," in *29th Tonmeisteragung - VDT International Convention*, Cologne, 2016.
- [25] F. Zotter and M. Frank, "Signal Flow and Effects in Ambisonic Productions," in *Ambisonics - A Practical 3D Audio Theory for Recording, Studio Production, Sound Reinforcement, and Virtual Reality*, Graz, Springer Open, 2019, pp. 99 - 130.
- [26] M. C. Green and D. Murphy, Composers, *Eigenscape* - <https://zenodo.org/record/1012809>. [Sound Recording]. University of York. 2017.
- [27] A. Rahman, "Portable Ambisonic Impulse Response System (P.A.I.R.S)," March 2017. [Online]. Available: <https://cnmat.berkeley.edu/projects/pairs>. [Accessed 17 June 2019].
- [28] A. Pérez-López and J. De Muynke, "Ambisonics Directional Room Impulse Response as a New Convention of the Spatially Oriented Format for Acoustics," in *144th AES Convention*, Milan, 2018.
- [29] Audio Engineering Society, "AES69: AES standard for file exchange - Spatial acoustic data file format," 2015.
- [30] IEEE, *Std 1722 (IEEE Standard for a Transport Protocol for Time-Sensitive Applications in Bridged Local Area Networks)*.
- [31] IEEE, *Std 1722.1 (IEEE Standard for Device Discovery, Connection Management, and Control Protocol for IEEE 1722 Based Devices)*.
- [32] XMOS, "XMOS Microcontrollers," [Online]. Available: <https://www.xmos.com/developer/products/silicon>. [Accessed 22 June 2019].
- [33] SHARC, "SHARC ADSP21489," [Online]. Available: <http://www.analog.com/en/products/audio-video/audio-signal-processors/sharc/adsp-21489.html>. [Accessed 22 June 2019].
- [34] "IEEE Standard for Local and metropolitan area networks--Bridges and Bridged Networks," *IEEE Std 802.1Q-2014 (Revision of IEEE Std 802.1Q-2011)*, pp. 1-1832, 12 2014.
- [35] "IEEE Standard for Local and metropolitan area networks--Audio Video Bridging (AVB) Systems," *IEEE Std 802.1BA-2011*, pp. 1-45, 9 2011.
- [36] "IEEE Standard for Local and Metropolitan Area Networks - Timing and Synchronization for Time-Sensitive Applications in Bridged Local Area Networks," *IEEE Std 802.1AS-2011*, pp. 1-292, 3 2011.
- [37] D. Kostadinov, J. D. Reiss and V. Mladenov, "Evaluation of distance based amplitude panning for spatial audio," in *ICASSP*, 2010.