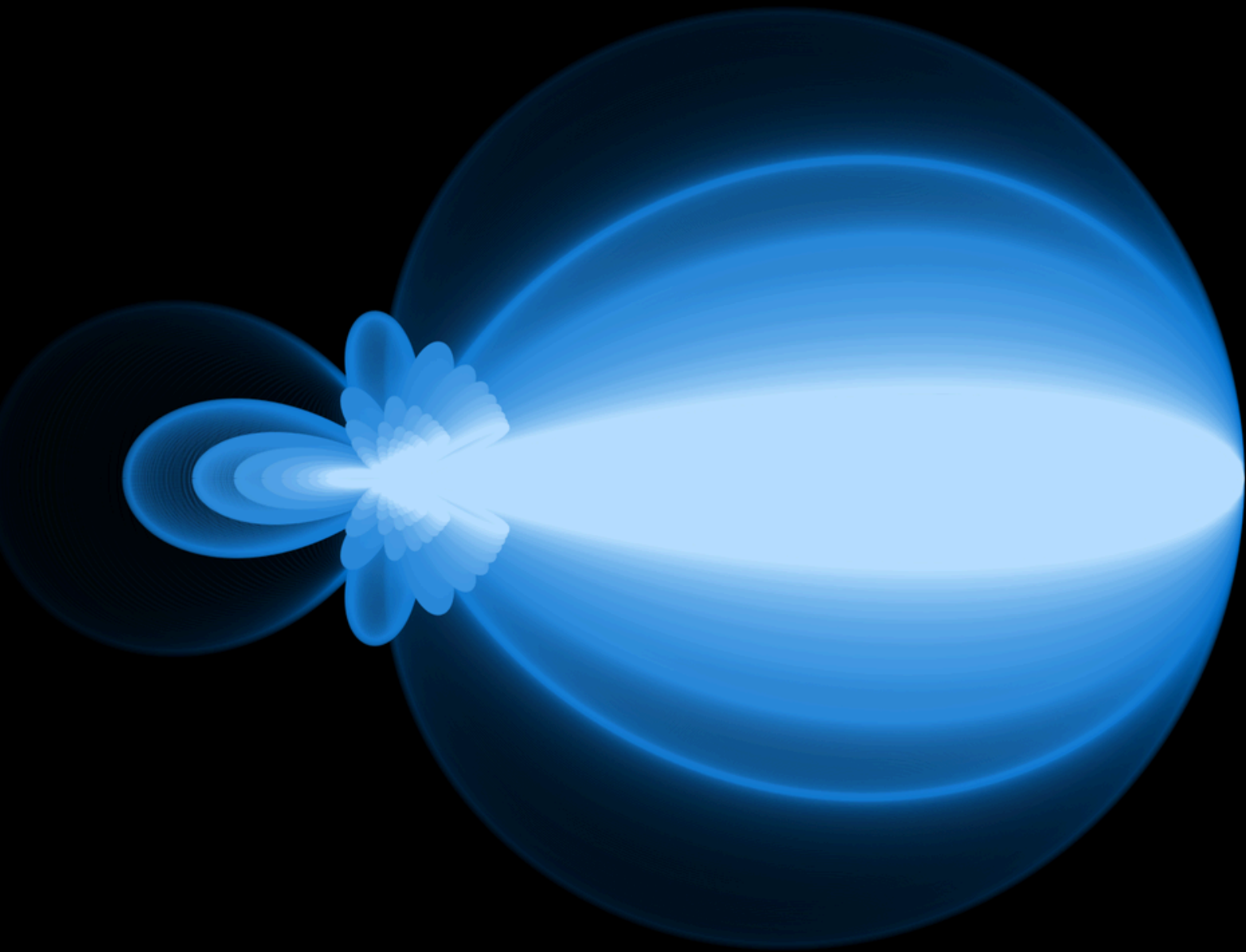


Getting to Know **Ambisonics**

**A Practical Guide for Sound
Engineers, Sound Designers and
Artists**



Peter Stitt



Getting to Know Ambisonics

A Practical Guide for Sound Engineers, Sound Designers and Artists

Version 1.0 (28th October 2026)

© 2026 Peter Stitt.

You are welcome to copy and share this eBook free of charge, provided that it is distributed in its complete and unmodified form. It may not be sold, modified, or incorporated into another publication or commercial product without prior written permission, except where permitted by law.

Third-party material, trademarks, product names and company names remain subject to the rights of their respective owners. Their inclusion does not imply endorsement or affiliation.

Every effort has been made to ensure that the information in this eBook is accurate at the stated version date. Software, platforms, formats and technical specifications may change, so current requirements should be checked with the relevant provider where appropriate. If you notice outdated information, please get in touch via the email below.

Published by SSA Plugins

The current version of this eBook is always available at

www.ssa-plugins.com/ebook

peter@ssa-plugins.com

Contents

Contents	2
1 Introduction	4
2 Ambisonics: Explained	8
2.1 Overview	8
2.2 Ambisonic Order	11
2.3 Ambisonic Panning and Ambisonic Microphones	16
2.4 Virtual Microphones (Beamforming)	18
2.5 Ambisonic Processing	19
2.6 Ambisonic Decoding	27
3 Binaural and Ambisonics	33
3.1 Binaural Hearing	33
3.2 Ambisonic-to-Binaural Decoding	39
4 Tips for Working with Ambisonics	45
4.1 Setup and Signal Integrity	45
4.2 Monitoring and Mix-Checking	49
4.3 Mixing Considerations	51
4.4 Creative Considerations	52
Ambisonics Quick Reference	56
A DAW Setup and Example Projects	58

<i>CONTENTS</i>	3
B Glossary	62
Bibliography	71

1 Introduction

Spatial audio gives you the ability to transport listeners to new sonic worlds. With so many different immersive formats, though, it's easy to get overwhelmed. This eBook focuses on Ambisonics, which is a flexible, well-researched format with a history going back to the 1970s. You will learn about what Ambisonics is, some of the theory behind it (with **almost** no maths¹!), where it sits in the immersive audio landscape, how it relates to how we actually listen to sounds, and how to use it in practice.

Why I Wrote This eBook

Since releasing the [aX Plugins](#) in 2018, I have been asked a lot of questions from people who are excited about Ambisonics but might be intimidated by it. I decided that it would be worth putting what I've learned over the years into one place so that people could get a useful introduction to Ambisonics, along with tips to avoid the mistakes I made! I want to make the concepts behind Ambisonics as accessible to as many people as possible without getting bogged down in overly academic language².

¹You might have read other guides to Ambisonics aimed at an academic audience. These are usually full of equations explaining the key concepts. Unfortunately, as great as these guides are, this can scare a lot of people off. The truth is, you don't need to know the equations to produce amazing sonic compositions with Ambisonics. This eBook avoids the maths and sticks to higher-level concepts and practical things that will help get you going.

²To satisfy my inner researcher, I have included references to scientific journal articles that support the points that I make. There is no need to read them to understand and work with Ambisonics, but interested readers who want to go deeper should find a lot of additional value in them.

Why Use Ambisonics

Ambisonics is not a speaker format. It is a representation of the sound field itself. Instead of storing speakers, like channel-based audio, it stores where sound energy exists in space. The spatial structure of the sound field is encoded directly in the audio itself, unlike object-based systems that store spatial information in metadata. Because of this, Ambisonics is not tied to any particular loudspeaker layout. The same Ambisonic signal can be decoded to stereo, 5.1, 7.1, binaural, or large-scale loudspeaker arrays.

This separation between representation and reproduction makes Ambisonics a powerful format for immersive workflows. **You can mix once, and decide later how it will be rendered.** And because the spatial data is stored in the Ambisonic audio itself, it can always be re-decoded later.

Ambisonics is increasingly used and supported in a range of professional contexts: spatial audio for games and VR/AR, film and documentary sound design, immersive video, museum and gallery installations, and architectural acoustics. It is also supported within modern immersive-audio standards and codecs including ADM, IAMF/Eclipsa and Apple's APAC. Ambisonics sits alongside channel- and object-based immersive formats rather than competing with them. It is particularly well-suited to recording-based workflows, VR content, and any project where the final playback format is not yet fixed at mix time. Learning Ambisonics opens up workflows and opportunities that neither stereo nor traditional surround can offer.

Tools You Can Use (Optional)

To make Ambisonics more accessible, I developed the aX Plugins, which you will see referenced throughout this eBook. The aX Plugins allow you to convert mono or stereo sources to Ambisonics, process the scene, and decode to binaural headphone mixes. They are also compatible with the signals from the expanding number of Ambisonic microphones on the market.

If you want to work along with the exercises, you can download a 15-day free trial of all the aX Plugins from www.ssa-plugins.com/downloads. To follow along with the exercises, download the aXBundle which provides the highest spatial resolution processing. The exercises aim to make some of the more abstract elements of Ambisonics into concrete examples you can hear. Hands-on work will also highlight the many workflow parallels and similarities to working with stereo.

Everything you learn here applies to Ambisonics regardless of which tools you end up working with. The goal is to give you a solid, practical understanding of the format itself that will serve you, whatever your setup.

How This eBook Is Structured

Whether you want to learn the bare minimum and just get started, or you want to explore more deeply, this eBook has you covered. Chapter 2 and chapter 3 will introduce the main things you need to know, along with some exercises using the provided projects, to make the concepts more concrete. Then, in chapter 4, I'll outline some tips and pitfalls to consider when working with Ambisonics that should make your own projects more stress-free.

In appendix A you can download the DAW projects needed for the exercises

and find a brief overview of how the main DAWs handle Ambisonics.

Tip

You are going to read phrases like "just like for stereo" more than once in this eBook. I've tried to keep it to a minimum but I really wanted to emphasise that even though you might not have come across Ambisonics before, you probably already know enough from working with stereo to include it in your workflow quite easily.

2 Ambisonics: Explained

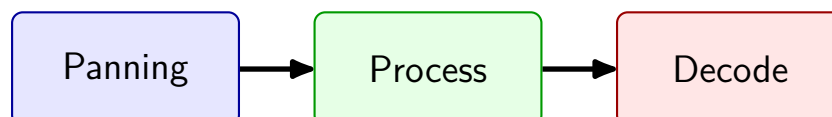
Goals

By the end of this chapter you will understand:

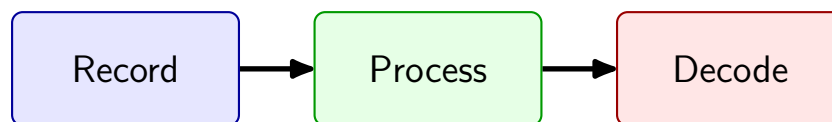
- Ambisonic signal flows.
- Ambisonic order.
- Ambisonics as a scene-based representation.
- Different ways to create Ambisonic signals.
- How to process Ambisonic signals.
- How to convert from Ambisonics to signals you can listen to.

2.1 Overview

Ambisonics is a spatial audio method developed in the 1970s by Oxford mathematician Michael Gerzon [1]. Ambisonics differs from many other immersive and surround formats because it is not tied to a particular speaker layout. It is an intermediate format that is not meant to be listened to directly. It must first be decoded for your playback system. The standard signal flow is:



Or if you're working with an Ambisonic microphone:

**Tip**

The most important shift when learning Ambisonics: think in scenes, not channels. In stereo or 5.1, each channel maps directly to a speaker. In Ambisonics, all channels together represent a single spatial sound field, meaning channels must always be processed together as a unit. You cannot treat them independently the way you might with a multi-track session. It is also why Ambisonics is not tied to any one playback format: the same scene can be decoded later to stereo, binaural, or a loudspeaker array.

In its simplest version, called first-order Ambisonics (1OA), it is essentially an extension of the stereo mid-side technique. Mid-side uses an omnidirectional¹ microphone along with a figure-8 one that captures the left-right sound field. 1OA adds figure-8s for front-back and up-down, giving a total of four channels. Figure 2.1 (left) shows these polar patterns. In theory, you can natively record FOA using an omnidirectional microphone coincident with three figure-8 mics, each pointing in the x, y and z axes. Figure 2.1 (right) shows the 3D directivity patterns for 1OA in 3D.

Higher Order Ambisonics (HOA) is when more channels are added to the signal to obtain higher spatial resolution. The directivity patterns of these channels, known as spherical harmonics, become much more complex than the figure-8 patterns of FOA. You can see them in fig. 2.2. In fact, spherical harmonics become complex enough that no practical microphone can capture

¹Or sometimes a cardioid microphone. The original mid-side used an omnidirectional mid channel [2].

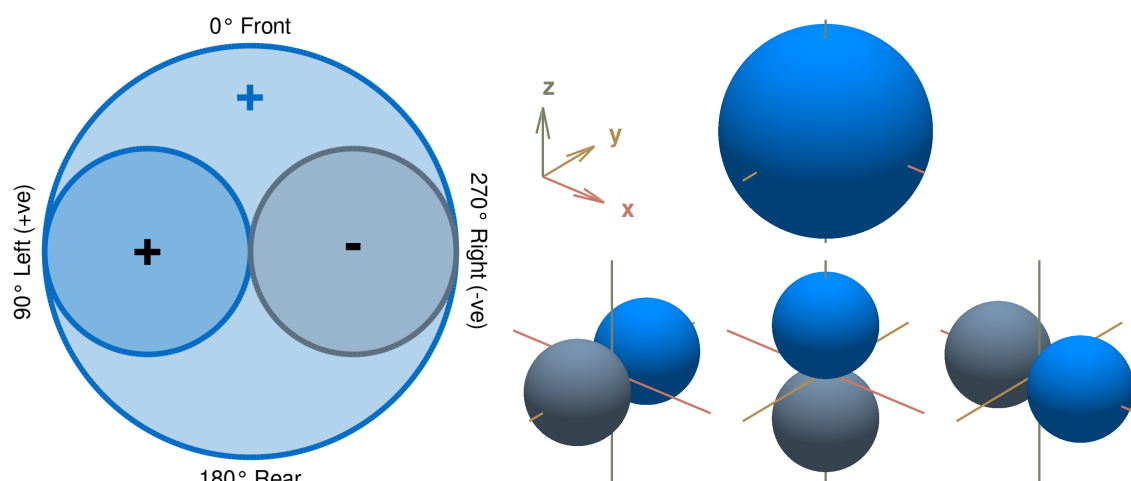


Figure 2.1: (Left) Mid-Side microphone polar patterns. The Mid channel is captured by an omnidirectional microphone. The Side channel is captured by a figure-8 microphone pointing left-right. (Right) The directivity patterns for first-order Ambisonics. The omnidirectional directivity pattern is on the top row. The X, Y and Z directivity patterns are aligned with the corresponding axes. The blue indicates positive polarity. The grey indicates a negative polarity.

them directly.

Just as combining the omni and figure-8 patterns in mid-side produces a cardioid, combining HOA channels with different weights produces increasingly precise directional patterns, leading to higher spatial resolution. We will explore exactly what "order" means in practice in more detail in section 2.2. For now, think of it as adding more coincident microphones to record the scene, albeit microphones with impossible directivity patterns.

Once you have your Ambisonic signal, whether recorded or panned², it needs to be decoded before you can hear it as a correct spatial scene. This is discussed more in section 2.6. The decoding process uses information about your playback system layout and converts to signals that will try to reproduce the intended spatial properties. **That means you can mix for Ambisonics and monitor over your own speakers or over headphones, send the**

²More correctly *encoded*, but panning is a more familiar concept to most people used to working with more traditional formats, so we'll stick with that.

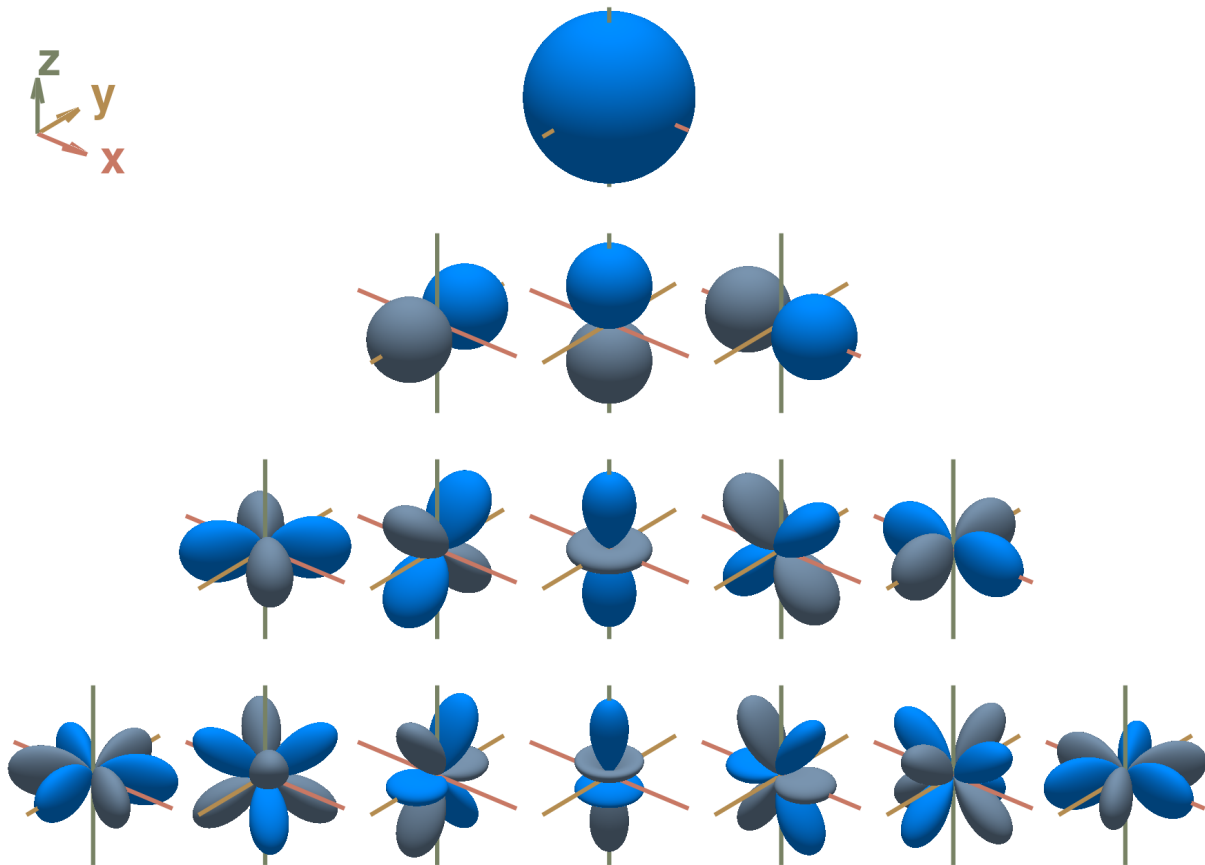


Figure 2.2: The directivity patterns of the 16 third-order Ambisonics channels laid out in hierarchical format. Each row contains all of the directivity patterns required for a new Ambisonic order i.e. rows 1 and 2 are first-order, while rows 1, 2, and 3 form a full second-order set. The blue indicates positive polarity. The grey indicates a negative polarity.

Ambisonic signals to someone and then they can listen to it on their own system.

2.2 Ambisonic Order

Ambisonic order is one of the most important concepts to understand because it is one of the main factors that determines **how precisely sounds can be localised in space**. It is also one of the first decisions you will need to make when starting an Ambisonic project.

If you take a simple Ambisonic scene and listen to it at first order, you will

often notice that sources feel somewhat diffuse. Movement is still apparent, but positions might not feel sharply defined. As you increase the order, the same scene begins to tighten: sources become more focused, their positions more stable, and their movement easier to follow. The difference between first-, second- and third-order is usually quite striking, while improvements beyond that tend to be more gradual [4].

What you are hearing is a change in spatial resolution. **The order of a signal defines the spatial resolution of the sound scene.** As the order increases so does the resolution. The trade-off is that higher orders require more channels. More channels means more processing and more storage is required.

At first-order only 4 channels are required, second-order needs 9 channels and third-order needs 16 channels. For an Ambisonic order N a total of $(N + 1)^2$ channels are required³. You can see how the number of channels starts to increase steeply with order. Table 2.1 shows the number of channels required for common orders up to tenth-order.

Ambisonic order	No. channels $(N + 1)^2$
1	4
2	9
3	16
7	64
10	121

Table 2.1: The number of channels needed to represent an Ambisonic sound scene for commonly used orders from 1 to 10.

³There it is! The only formula in this eBook! I did promise to keep it light.

Tip

Downmixing from a higher Ambisonic order to a lower one is trivial. All you need to do is drop the channels of higher orders. No reordering, matrixing, or fancy algorithms needed. If you have a third-order signal this can become first-order by just keeping the first four channels in the signal. If you want to downmix to second-order then just keep the first nine channels. Easy.

What Order Do I Need?

So, what order do you actually need to use? That's a question without a definitive answer because it depends on your end goal. If your final delivery format is third-order Ambisonics then you could very reasonably work only in third-order. I'd recommend working at the highest order that is useful for your sources, delivery target and practical constraints. When practical, retaining a higher-order working master can help future-proof the project. That way any improvements on the playback side that increase the decoding order don't lead to the need to rework the whole project.

If you've got a signal of, say, third-order and you need to provide a first-order version then the "downmix" is trivial: all you need to do is take the first four channels of the signal and you've got a first-order downmix. Now let's say someone asks for a seventh-order version (which has 64 channels). Your third-order version is lacking in information. At this point you have the option of remixing the project again, which means ensuring your DAW has all the correct routing. Far from impossible but definitely time-consuming.

If you are working exclusively with recordings from, say, a first-order microphone then you don't need to work with higher orders for your processing. In

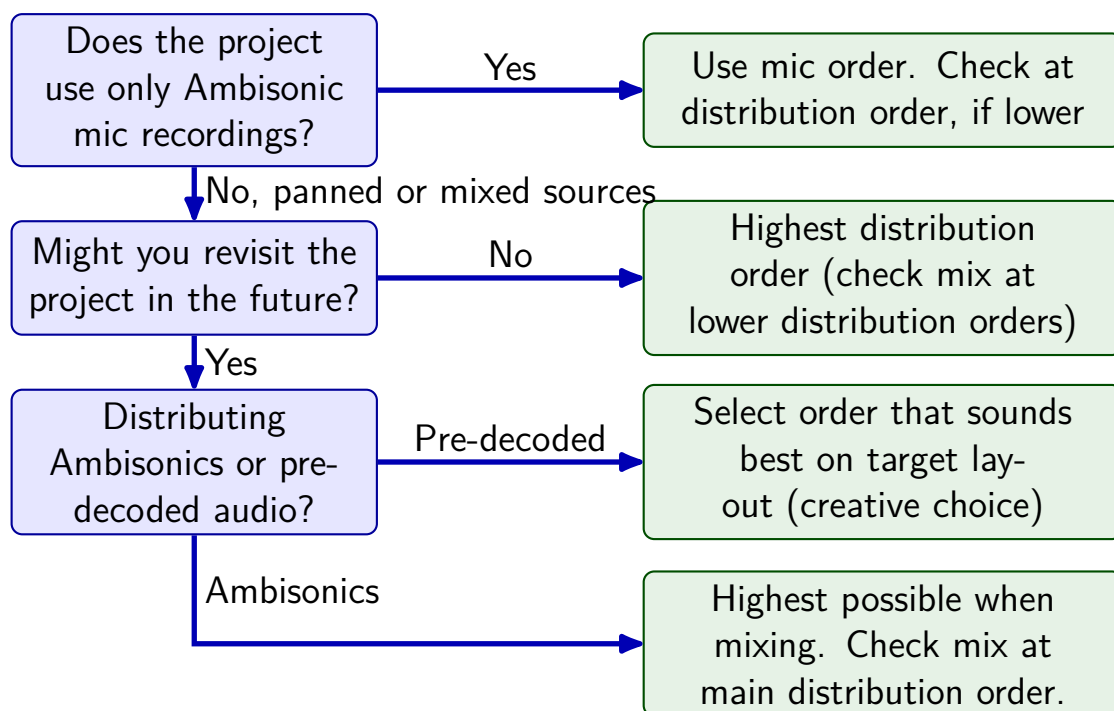


Figure 2.3: Decision tree for selecting Ambisonic order for a new project.

fact, **you shouldn't process at higher orders than the signal**. Processing at a higher order than your source signal adds channels that contain no real information, wastes CPU, and can introduce artefacts at the decoder. The [aX Plugins](#) will internally adapt to work with the signal you feed them, to save CPU without having to manually select the processing order. But it's still better to never process or decode a signal with tools specifically set to a higher order.

The decision tree in fig. 2.3 can help you decide what order you should be working at. Table 2.2 provides some specific references for common distribution options.

Tip

The most common orders are first-, second-, third-, and seventh-order. There are affordable first- and second-order microphones available that make working at these orders attractive, but third-order is also a sort

of sweet-spot between number of channels and overall quality. Seventh-order was, for a long time, the maximum order possible due to most DAWs limiting tracks to 64 channels. That changed with the release of Reaper 7 which allows 128 channels per track, opening up the possibility for tenth-order processing. So far, if you want to work at the highest possible spatial resolution, Reaper is currently the only game in town.

Project/Delivery Type	Working/Delivery Order
Binaural production	3OA or higher if resources allow
YouTube	1OA for standard Ambisonic spatial audio ⁴ . Eclipsa Audio is also supported for eligible partners ⁵ .
VR/AR (real-time rendering)	Often 1OA to 3OA depending on renderer
1OA microphone recording only	1OA
HOA microphone recording	Match microphone order
Archival/future reuse	Keep the higher working order
Eclipsa/IAMF	Current open reference renderers support up to 4OA ⁶
Apple Positional Audio Codec (APAC)	Profile 5 supports up to 3OA ⁷

Table 2.2: Quick reference: Ambisonic working and delivery order by project type.

Exercise

Open the **OrderComparison** project. Open the aXPanner from the FX chain on the panner bus. The automation is set to pan the sound around you from the front, going right, back, left, then front.

⁴<https://support.google.com/youtube/answer/6395969?hl=en>

⁵<https://support.google.com/youtube/answer/1722171?hl=en>

⁶<https://github.com/AOMediaCodec/oar>

⁷<https://developer.apple.com/av-foundation/Apple-Positional-Audio-Codec.pdf>

There are three different sounds routed to the panner, which you can solo individually. This will let you hear how increasing the order impacts sounds with different frequency contents and signal characters.

While playing back the signal, try changing the **Order** parameter to 1. Listen closely to the spatial impression and timbre of the binaural signal.

Now try gradually increasing the **Order**. You should see the change in order reflected on the bottom left corner of the aXMonitor plugin UI.

Can you hear a difference as the order increases? Most people will hear a big increase when going from first-order to third-order. After that the improvements are less dramatic depending on how well the decoder HRTF matches your own ears. With a good matching HRTF you will hear improvements for each increase in order.

2.3 Ambisonic Panning and Ambisonic Microphones

Just like with stereo or 5.1 surround, you can take a mono signal and convert it to Ambisonics using a panner, like [aXPanner](#). Alternatively, you can also use a microphone to capture Ambisonic signals directly [5].

Recording an audio scene using an Ambisonics microphone lets you capture the full 360° experience. You get the direct sound, room reflections, background sounds, source motion all in the same signal. FOA mics use four closely spaced capsules in a tetrahedral arrangement. HOA mics have more capsules and are therefore more expensive, but there are good options available for those who need the additional spatial resolution they provide. In general, microphone arrays cannot record natively to Ambisonics. Instead, they record

to what is known as A-format, which is simply the raw audio from the microphone capsules. Manufacturers will provide some tool for converting from the raw A-format signals to what is known as B-format. B-format is simply the microphone workflow way of saying the signal is now Ambisonic. Usually, A-to-B conversion takes the form of an audio plugin that performs some specialised processing. Once the signal has been converted, you treat it as you would any other Ambisonic signal.

Tip

Always keep the raw microphone recordings, along with the HOA conversions. This way, if there is an update to the tools that leads to improved conversion, you can apply it later. And by keeping the HOA conversions you are sure you have the tracks in case the conversion plugin ever gets discontinued or becomes incompatible with newer systems.

Ambisonic panning takes a signal and applies gains to it to generate the Ambisonic signal. Sources can be panned in azimuth (around the listener) and elevation (above or below the listener). Panning is limited in order only by the software channel counts or the computer's CPU. For example, Reaper 7 expanded its tracks to allow 128 channels, which supports high resolution tenth-order Ambisonics. When panning signals, however, there is no ambience or room effect. These have to be simulated to create an even more realistic sound scene.

The choice between microphone and panning is completely project dependent but most projects can benefit from a mix of the two. Ambient sounds recorded with a microphone can be very immersive because they contain a lot of complexity thanks to the correct reflections and reverberation. Panned sounds can be much more precise and be used for additional spatial

clarity, control or a sense of hyper-realism. However, if you are creating scenes with mixed orders (e.g. first-order ambience, third-order spot mics) then you should read section [2.6](#).

Warning!

You should not listen directly to an Ambisonics signal. Until it has been decoded, Ambisonics does not contain any direct speaker feeds. Section [2.6](#) and chapter [3](#) give more detail about the decoding process.

Tip

Record a scene with an Ambisonic microphone and spot mics. Pan the spot mics using HOA while matching the direction of the sound in the Ambisonic microphone signal. Now you have direct sound that is more precise and localisable, while retaining the ambience that will reinforce immersion in the scene.

Summary

In short, microphones give you immersive realism; panning gives you precision. Many projects could benefit from a thoughtful mix of both.

2.4 Virtual Microphones (Beamforming)

It is possible to take the Ambisonic scene and exploit the spatial information it contains to generate microphone signals. This is also known as beamforming.

By combining all of the channels with different weights we can create new signals with a new directivity pattern. This is exactly the same as adding a figure-8 pattern to an omnidirectional one to get a cardioid pattern. The same principle scales up so more channels leads to more complex patterns, as well as far greater directional precision.

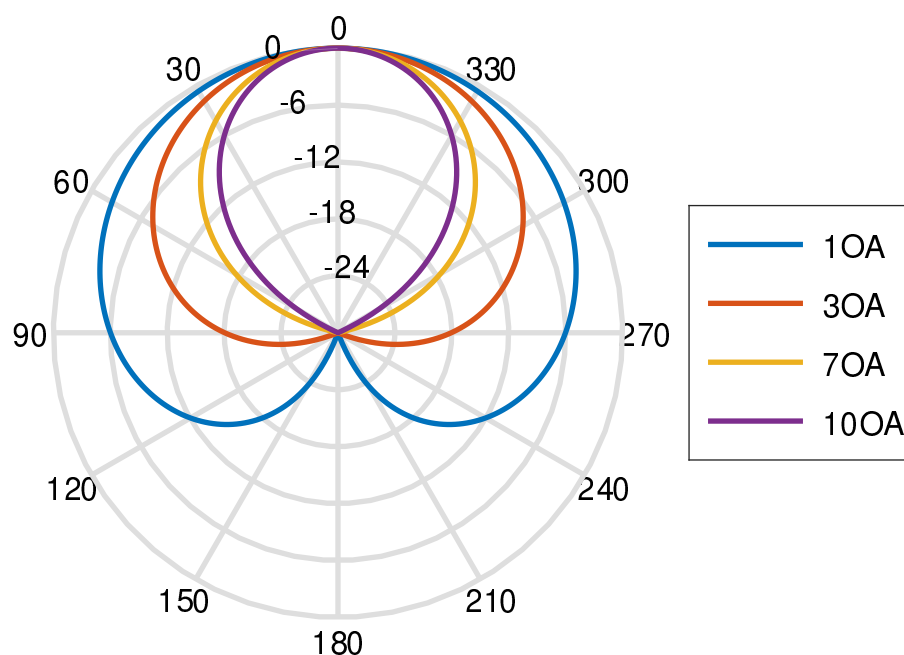


Figure 2.4: The polar patterns for cardioid directivity patterns of increasing order. These patterns are created via weighted combinations of spherical harmonics. Increasing order produces more directive patterns.

With HOA signals we can create cardioid or higher order cardioid directivity patterns. Figure 2.4 shows the polar pattern of cardioid directivity patterns from first-order to tenth-order. Each increase in order produces a narrower, more focussed pattern. The narrowing provides an additional 6 dB of attenuation at $\pm 90^\circ$, meaning sounds from the sides are attenuated more, so the pattern focuses more on what is in front. This is exactly what “higher spatial resolution” means in practice: sounds can be more precisely localised in the scene.

2.5 Ambisonic Processing

Once your signal is converted to Ambisonics you have several options for processing. Just about any signal processing you can do in stereo can also

be applied to Ambisonics. Conventional processors can usually be applied safely if they apply identical, linked processing to every Ambisonic channel. Processors specifically designed for Ambisonics may deliberately transform or mix the channels while preserving, or intentionally modifying, the represented sound field.

Ambisonics opens up creative possibilities that simply don't exist in stereo, because space becomes a dimension you can easily manipulate and use alongside frequency and time.

Let's explore the different types of effects you can apply, when, and what to watch out for.

Dynamic Range Processing

Dynamic range processing can be applied easily, as long as it respects the phase and gain relationships between the channels. This kind of effect essentially uses a mono signal derived from the Ambisonic signal to feed a "side-chain" of the processor. This could be the omni-directional channel or could be a virtual microphone signal pointing in a certain direction.

If you are panning mono sources, it might be better to apply dynamic range processing before you convert your signal to Ambisonics. This way only one or two channels are processed at a time. But, just like for stereo, applying the compression to a mixture of Ambisonic tracks can help to glue them all together.

Of course, if you recorded natively in Ambisonics then you'll need Ambisonics-compatible dynamic range processing no matter what.

Exercise

Open the **DynamicRange** project. Open aXCompressor and set a moderate ratio (3:1) with medium attack and release times.

Increase the Focus of the virtual microphone to a value greater than 0% (omnidirectional). This uses a directional signal derived from the sound field to drive the compression, rather than the full Ambisonic signal. Point the virtual microphone toward the main source in your scene and engage the compressor. Notice how the compression responds primarily to that source while excluding the rest of the scene, particularly for the highest virtual microphone focus.

Now rotate the virtual microphone away from the source. The compression behaviour changes completely: the compressor's gain reduction is now reacting to a different part of the scene entirely.

You can do the same for aXGate.

This kind of spatial sidechain control is simply not possible in stereo. It gives you precise dynamic control over your scene without breaking the spatial relationships between channels.

Equalisation

Equalisation is as straightforward in Ambisonics as it is in stereo. In general, an Ambisonics EQ will have a selection of filter types, gains, Q-values, and frequencies. Using an Ambisonic EQ is much the same as using a mono or stereo one *as long as the gain and phase changes are constant across all channels*. [aXEqualizer](#) was designed to work this way, as well as using efficient processing methods to handle the large number of channels.

As with dynamic range processing, EQ is best applied to either Ambisonic

microphone recordings or to buses. If you want to apply independent EQ curves to several mono signals that you're panning then you can consider applying the EQ to each source before panning to Ambisonics. This will save you CPU: The EQ and Ambisonic panning are linear, so you'll end up with the same signal no matter what order you do them in.

Exercise

Open the project of your choice and insert aXEqualizer on the Ambisonic bus, or after an instance of aXPanner. Apply a broad high-shelf boost of around 6 dB above 8 kHz. Listen carefully to the binaural output through aXMonitor. The tonal change should be clear but the spatial image should remain stable and well-defined.

Now try a more dramatic change: apply a deep narrow cut (high Q, around -12 dB) somewhere in the 2-4 kHz range. Again, the tonal impact is obvious but the spatial image holds together.

This is aXEqualizer applying identical processing to all channels simultaneously, preserving the gain and phase relationships that encode the spatial information. As you experiment with different filter settings, notice that no matter how dramatic the EQ curve, sources stay where you placed them in the scene.

Rotation

A key advantage of Ambisonics is that it's possible to easily apply rotation to the whole sound scene.

Rotation can be used to align recorded sound fields, add dynamic movement to the scene or, perhaps most powerfully, to connect with a head-tracker. Adding head-tracking means the scene will be rotated in the opposite direction

to how the listener's head moves. The result is a scene where sources appear correctly fixed in space, rather than moving with the head. **This can help to improve externalisation and realism** [6, 7].

Rotation is usually applied in terms of yaw, pitch and roll, i.e. around the z-, y- and x-axis respectively. The order of the rotation matters, so applying yaw-pitch-roll isn't the same as pitch-roll-yaw. Most of the time the yaw-pitch-roll convention is used. If you're just applying a rotation effect, like [aXRotate](#), then you don't need to worry about this order. This might become a concern if you're using a head-tracker. In that case you'll want to verify the convention it is using matches the rotation effect.

Tip

The final project delivery format does not always support head tracking. There is research [8] that suggests that adding some small oscillations to the sound field can help to improve spatial impression and front-back differentiation when head tracking isn't available.

[aXRotate](#) allows you to add **Micro-movements** that do exactly this. You can choose the amplitude of the motion until you find something that works for your scene. The ideal is to add just enough that it improves the spatial impression without being heard as motion of the sound field.

Exercise

In the **MicroMovements** project, try adjusting the amplitude of the micro-movements in [aXRotate](#) until the movements become audible to you. This will depend on your own natural sensitivity, but also on the complexity of the sound scene and the position of the sounds in it. Look at the [aXMeter](#) with micro-movements turned on and off to see that it's actually applying the rotation even if you cannot hear it.

Try applying a yaw rotation of varying values to hear if your sensitivity to different micro-movement amplitudes changes. You will probably find that you can hear smaller changes when the sound is directly in front of you than when it is to your side. This is because the minimum audible angle that we can hear changes depending on the source direction [9].

Spatial Delay

An Ambisonic delay has all of the same parameters you probably already know: delay time, feedback, gain. With [aXDelay](#) things get even more interesting because you can also control the spatial placement of each delay-line by applying a rotation to each. Think of it as a ping-pong delay where you have full three-dimensional control over where each echo appears in the scene.

This opens up significant creative possibilities. For example, you can keep the delay times short and distribute them evenly around the listener, then play with the feedback and gain to get a subtle room-type effect. Or you can use long delay times and strategically placed directions to get striking special effects to make a sound really grab the listener.

Exercise

Open the **OrderComparison** project and insert [aXDelay](#) after the instance of [aXPanner](#). Start with a single delay line: set a delay time of around 300-500 ms, moderate feedback, and place the delay at 90°. Play back the signal and notice how the echo appears clearly separated from the dry source in space.

Now add a second delay line at a shorter time (around, say, 150 ms) and place it at -90° to one side. You now have echoes arriving from different

directions at different times, with full control of their position.

Finally, try reducing both delay times to under 30-50 ms and distributing three or four delay lines evenly around the listener with low feedback and moderate gain. This starts to approximate a simple room effect, mimicking early reflections. Adjust the gains until it blends naturally with the dry signal.

Notice how much spatial character you can add without any dedicated reverb plugin.

Reverb and Adding Space

One of the most interesting and creative elements of immersive audio is being able to create a sense of space for the listener. If you record with an Ambisonic microphone then you get the space for free in the recording. Panned audio is different because it has direction but is not placed in an acoustic space.

A mono source can be convolved with an Ambisonic room impulse response to place it within the captured acoustic space. The convolver needs to support the required multichannel Ambisonic output. The OpenAir library from the University of York contains a selection of free Ambisonic impulse responses from a variety of different spaces⁸. Noise Makers have a 7OA convolution reverb with built-in impulse responses⁹.

The direction of the direct sound is determined by the source direction captured in the impulse response. You can adjust the direction by applying rotation which will let you move the source to come from your preferred direction. Just be aware that you will also be rotating the room with it. For

⁸www.openairlib.net

⁹www.noisemakers.fr/product/ambi-verb-hd/

creative mixes this is unlikely to pose a problem but if you're looking for physical exactness then you need to be aware of the limitations.

Exercise

In the **RoomAndEncoding** project there are two tracks. One is a completely dry signal panned to 10A. The other is the same signal convolved with a 10A impulse response. The panned signal has been positioned to approximately match the direction of the direct sound in the convolved response. Solo each of them to listen to the difference. It is clear that the reverberant signal has much richer acoustic and spatial information, while the panned signal's position might be slightly more precise. Un-solo both tracks and listen to the effect of changing the gain on the panned signal. As the gain increases the direct sound becomes clearer. Large increases can give a kind of hyper-real perceptual effect, with clearer direct sound than the acoustics themselves have. This kind of dry/reverberant processing opens up creative possibilities, as well as allowing a mix to ensure crucial content is clear for the listener. Try adjusting the levels until the two blend in a way that improves the clarity without sounding artificial.

Summary

Ambisonics can be processed using most of the same tools as for stereo or other formats, as long as the effect maintains the spatial cues encoded in the signal. The [aX Plugins](#) are all designed with this in mind. The efficiency with which Ambisonics can be rotated makes head-tracking during mixing and playback very attractive.

2.6 Ambisonic Decoding

Ambisonics has to be decoded to the final output format (stereo, binaural over headphones, 5.1, *anything else*).

With the right decoder, Ambisonics can be decoded in a way that physically recreates the target sound field [10]. It is *holophonic* (like a hologram, but for audio). However, the listening area covered by this holophonic region gets smaller as the frequency increases. Eventually the area ends up smaller than the size of a human head placed at the sweet spot. At this stage, decoder design often moves to other methods of optimisation, like psychoacoustics.

Decoder Methods

There are different decoding methods for Ambisonics. The main ones are all used to create a decoding matrix that mixes the Ambisonic channels with different weights to generate loudspeaker signals. If you're targeting a common loudspeaker layout, your decoder probably has a good preset already built in. Unless you have good reason not to, pick the one that works best for your target output format.

However, if you're working with non-standard layouts, you'll need to develop a custom decoder. The main decoding methods are SAD, AllRAD, EPAD, MAD, and numerically optimised decoders. You can read a short description of the main decoder types in the Glossary (see appendix B). The choice of the decoder is a very important one because it impacts the spatial impression, the timbre, the performance at off-centre positions, and the loudness.

For most practical work, AllRAD is a sensible default for irregular loudspeaker layouts. If loudness consistency across the listening space matters

more than maximum localisation precision, EPAD is worth considering. SAD and MAD are included here for completeness but are rarely the right choice for real-world layouts. Numerically optimised decoders can give excellent results when quality is the priority and you have time to develop a custom solution.

If you know your target layout, try testing different decoding methods until you find one that works best for that layout. There is no rule that a single decoder will always work best across layouts.

Of course, Ambisonics can also be decoded to binaural for immersive listening over headphones. You can read more about that in [chapter 3](#).

Decoder Optimisation

Most advanced Ambisonic decoders work in at least two decoding bands. In the low frequencies these decoders may use what is called a Basic decoding method. In a Basic decoder all of the loudspeakers can be active at once so that the interference of all the speakers can create the holophonic sound field. That means that even loudspeakers in the opposite direction of the intended source direction can be active.

Whenever the frequency reaches the point where the signal is no longer well reproduced some decoders will make the choice to switch over to a psychoacoustically optimised version. This is called $\max r_E$ decoding. The aim is to concentrate the energy of the loudspeakers in the direction of the sound source. Rather than trying to reproduce a physically accurate sound field, this method aims to produce loudspeaker signals that maximise psychoacoustic cues. This can help with reproducing more accurate level differences between the ears. We'll look at why this matters for localisation in [chapter 3](#).

A third decoding optimisation is known as *in-phase* decoding. This aims to

ensure that, unlike for a Basic decoder, the loudspeakers on the opposite side of the loudspeaker layout to the source are not active. This can be really useful when playing back a scene to a large audience or where not many listeners can be near the sweet-spot. The trade-off is that the source width is larger than for Basic or max r_E decoding.

Decoder optimisation is performed by applying gains to the different Ambisonics channels, though in practice they are usually merged with the decoding matrix. The optimisation gains depend on the decoding order. That means if you try to decode a first-order signal with a decoder configured for second-order or higher the decoding will be suboptimal. This usually means the sources will have larger source width or will feel less precise to localise.

Mixed Decoder Orders

So far we have assumed that all of the signals that need to be decoded are of the same order. However, sometimes you will be working with different orders in the same project: Imagine that you have recorded ambiences using a first-order Ambisonic microphone but you have mono spot mics that you want to encode to third-order for higher precision. That's totally possible.

You could just sum all of the signals and send it to the same decoder that's configured to decode to the highest order but generally Ambisonic decoders are optimised for a particular order. This signal flow is shown in the top part of [fig. 2.5](#). That means that the lower order signal will be non-optimally decoded if any of the optimisations from the last section are used. If your lower order signals are mostly background elements then this probably won't have any negative impact for the listeners.

The better option, however, is to use one decoder for each order and then

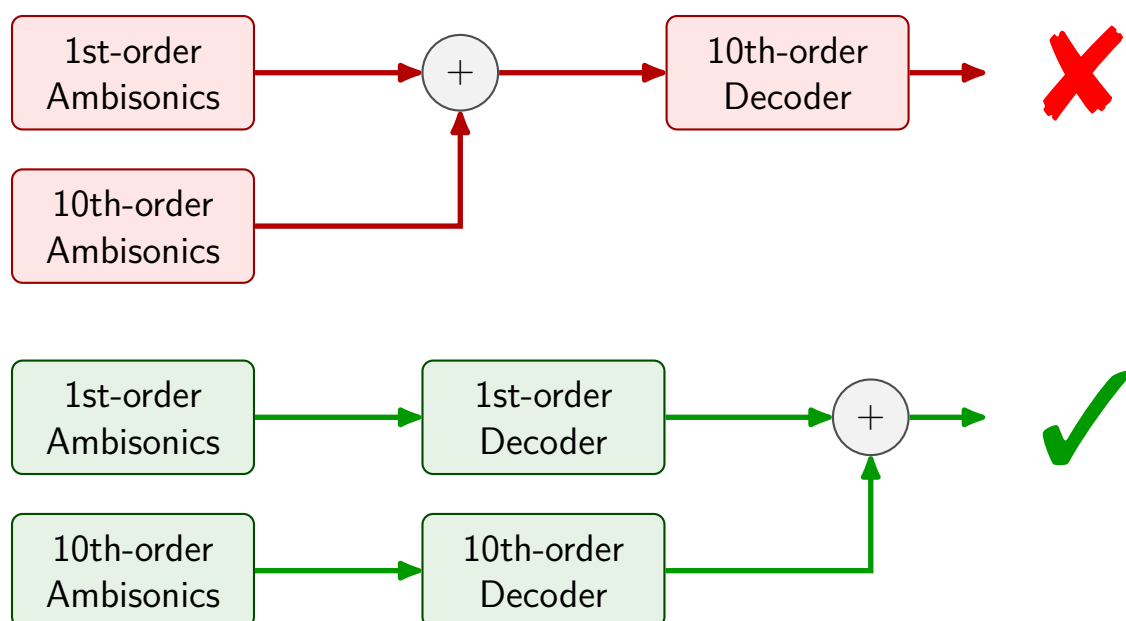


Figure 2.5: Example signal flows when working with Ambisonic signals of differing orders. It shows (top) signals being merged before being decoded together. This will produce suboptimal results for the first-order signal. (bottom) Decoding each order with appropriate decoders to the same output format before merging. This ensures that any order-dependent optimisation is applied correctly.

to mix the outputs of the decoders. Essentially, you will have a signal chain for each order you are using, keeping them separate. An example signal flow is shown in the bottom part of fig. 2.5. At the end of each signal chain you will have a decoder configured to the corresponding order. So, you would have one decoder bus that receives all of the, say, first-order signals and another for the third-order ones. Then, if you are decoding to binaural then you would simply send the stereo outputs of both decoders to your master stereo output.

Note that you might be distributing your final output as Ambisonics with decoding to happen later. In this case, you do not have the option to decode individual streams. I would recommend that you should keep your signal chains separate for the two orders until the final output, where you will simply mix them, similar to the top signal flow in fig. 2.5. This does mean that the lower order signals might be sub-optimally decoded but, as mentioned, if you have

kept the lower order sounds to background ones then it's less likely to be noticed by the listener.

Exercise

Open the **Mixed-Order** project.

There are different groups of tracks to demonstrate the difference between mixing orders before the decoder or using separate decoders for each order.

- The first sums a first-order signal with one encoded at the highest available in the DAW. This is the top signal flow in fig. 2.5.
- The second decodes each signal with the appropriate decoder, then sums it. This is the bottom signal flow in fig. 2.5.

Compare each of the mixes, paying particular attention to the matchbox signal.

For the first group, you should be able to hear a widening of the first-order source. The high frequency content of the signal is also attenuated. This is because the decoder optimisation (see section 2.6) applied to the first-order signal is configured for the higher order signal, and thus suboptimal. For the second group, the matchbox is reproduced with higher fidelity: the localisation is more precise and the high frequencies are retained.

Summary

Ambisonics is a scene-based audio format that can be created by recording with specific microphones or panning of mono or stereo signals. It can be processed to apply effects like rotation, spatial delay, compression and more. Ambisonics is not tied to any one playback format, with decoding

possible to stereo, 5.1, 7.1.4, binaural audio, and more.

3 Binaural and Ambisonics

Goals

In this chapter you will learn:

- The cues we use to localise sounds.
- How they are linked to Ambisonics-to-binaural decoders.
- How Ambisonic order impacts the quality of the psychoacoustic cue reconstruction.

One of the main ways of listening to Ambisonics is using a binaural decode - a process that converts from Ambisonics to binaural audio. This is so common because headphone-listening is one of the main ways people consume audio today, particularly immersive audio. Besides that, rendering HOA requires a potentially large number of loudspeakers, so **binaural rendering can be much more convenient**.

This chapter will go through the psychoacoustics of binaural listening before showing how a binaural decode of an Ambisonic signal can reproduce the desired psychoacoustic cues required for good spatial impression.

3.1 Binaural Hearing

Binaural cues are a major part of how we localise sounds in the spaces around us. Sounds arrive from a direction and, depending where they are, will reach the ears at different times. The signal will arrive at the nearer ear first and the further ear last. This produces what is called an Interaural Time Differ-

ence (ITD). For tonal components, this timing difference also produces an Interaural Phase Difference (IPD). The head will also block the sound from reaching the further ear. This acoustic shadowing creates what is known as an Interaural Level Difference (ILD). The ITD and ILD can be used to localise sound on a “cone-of-confusion”: a cone upon which the sound source could lie and produce the same cues. Without any additional information the brain has no way of knowing where exactly on this cone the sound is, resulting in some localisation ambiguity. This can commonly be in the form of front-back confusions, where a sound at the front is perceived as being behind the listener or vice-versa. In practice, most people perceive front-to-back confusions with binaural audio, rather than back-to-front.

One way to give the auditory system more information is to make small head movements [11]. The brain can then analyse how the ITD and ILD change as a result of these movements to get enough information to resolve a lot of the ambiguities.

In addition to dynamic cues, the auditory system evaluates elevation using spectral cues. The pinna (outer ear) will cause resonances and notches in the sound spectrum that depend on the position of the sound source. The auditory system is able to use these spectral cues to determine the elevation of a sound source. Because everyone’s ears are a different shape, the spectral modifications will be unique to each person. That’s why binaural audio made with someone else’s ears might not sound as good to you [12].

Taking everything together, these binaural and spectral cues are all included in the Head Related Transfer Function (HRTF) which provides a set of direction-dependent filters, one for each ear and each direction. Convolution of a sound with the left and right ear filters can give a spatial impression of a sound coming from that direction. Since these filters are person-dependent, the ideal

is to have your own personalised HRTF measured. Unfortunately, that isn't available to most people because it requires access to an anechoic chamber and specialised measurement equipment. There are also services that allow personalised HRTFs to be generated based on photographs or videos.

Tip

Getting personalised HRTFs isn't (yet) easy or cheap. Thankfully, the SOFA (Spatially Oriented Format for Acoustics) format^a was developed to allow easy exchange of different HRTFs [13]. There are now databases of HRTFs that you can test until you find one that sounds spatially and timbrally balanced to you.

^a[https://www.sofaconventions.org/mediawiki/index.php/SOFA_\(Spatially_Oriented_Format_for_Acoustics\)](https://www.sofaconventions.org/mediawiki/index.php/SOFA_(Spatially_Oriented_Format_for_Acoustics))

Exercise

First, download some .SOFA files from a database. A good one is the University of York's SADIE database: <https://www.york.ac.uk/sadie-project/database.html>. Download a few HRTF SOFA files from there and store them somewhere you can easily find later.

Next, open the aXMonitor plugin in the **OrderComparison** project. Click on the **HRTF** button, followed by **Load New .SOFA**. Select the HRTF that you'd like to listen through. The plugin will take a little time to process the HRTF to create the decoding matrix. How long depends on the density of the HRTF but it shouldn't be more than a few seconds. Once processing is finished you'll see the HRTF listed. Click on it so it's highlighted and then click **Use HRTF**.

Switch between HRTFs to hear the changes in timbre and spatial balance. Test different HRTFs until you find one that works well for you.

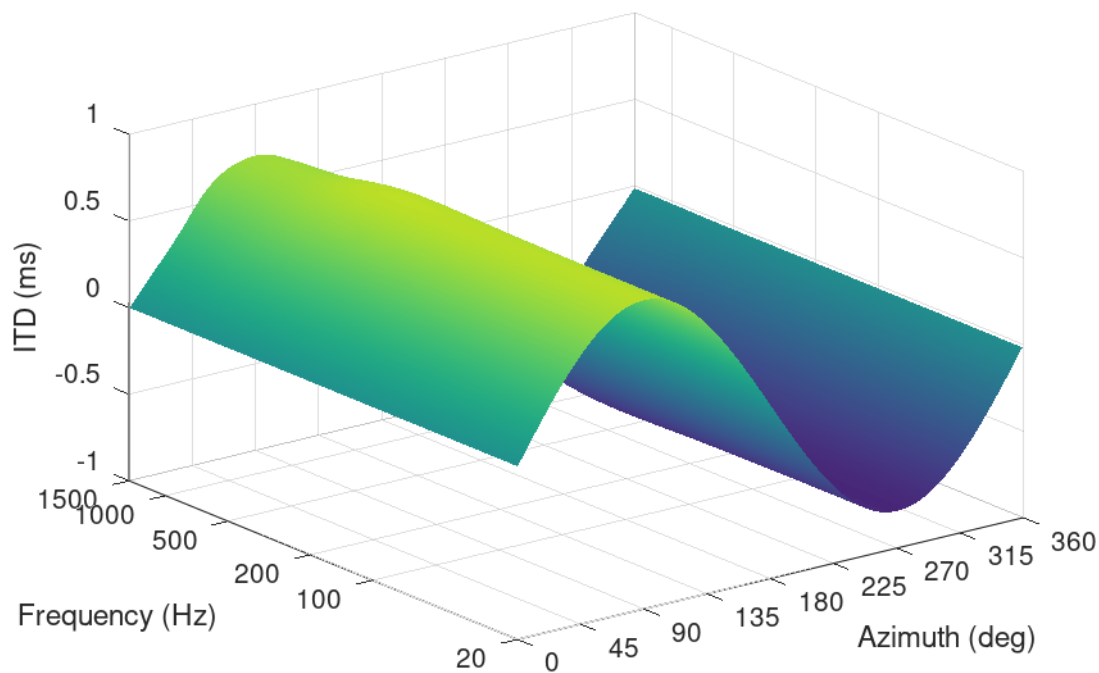


Figure 3.1: The ITD derived from the IPD, as a function of the source azimuth and frequency. It exhibits a broadly sinusoidal behaviour as the source moves around the listener.

Interaural Time/Phase Difference

ITD/IPD is the primary localisation cue used in the low frequency range. Figure 3.1 shows the ITD for a sound moving around the listener. It is broadly sinusoidal, with peaks to the left and right and zero to the front and back.

As the frequency of a signal increases its wavelength decreases. At the point where the wavelength is shorter than the distance between the two ears the phase becomes ambiguous and so IPD can no longer be used to reliably localise sounds.

Luckily, ILD steps in to pick up the slack.

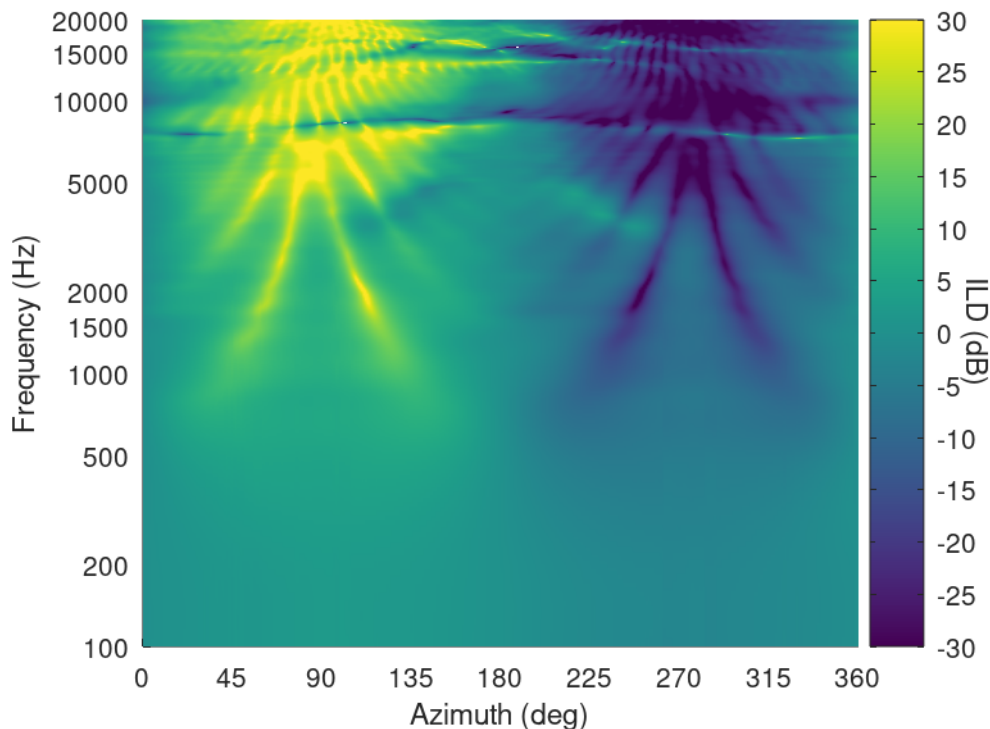


Figure 3.2: The ILD of a Neumann KU100 dummyhead HRTF [14] as a function of the source azimuth and frequency. The ILD is very low in the low-frequency range and gradually increases as the frequency increases.

Interaural Level Difference

ILD is frequency dependent, as shown in fig. 3.2. Long wavelength, low frequencies are able to wrap round the head with relatively little obstruction. This is the flat region at the bottom of the plot. As frequency increases the amount of scattering increases. The low ILD at low frequencies means that it is not a useful cue for the auditory system in this frequency range.

At higher frequencies the ILD increases and so does its usefulness as a localisation cue. You might also have noticed that ILD is not exactly as regular as the IPD. This is because of the way sound scatters around the head at different frequencies.

Duplex Theory

Combined with ITD/IPD at low frequency this leads to the Duplex theory of localisation [15] in which ITD/IPD dominates at low frequencies and ILD dominates at higher ones. The cutoff is at around 1.5 kHz, though there is a frequency range with some overlap. Broadband sounds will use some combination of the two in order to come up with a single judgement of the localisation. If the cues aren't coherent for a real sound source then this can lead to a blurry or unstable sense of source location. That's why it's important for any spatial audio system to try to recreate cues similar to those expected by the auditory system.

Spectral Cues

Spectral cues usually start around 4 or 5 kHz and are due to resonances in the outer ear. Figure 3.3 shows the transfer function for the left ear for different sound source elevations (-30° , 0° , 60°) coming directly from the front (azimuth = 0°). Notice how the peaks and notches vary and can get very deep. These spectral changes help the auditory system to work out the elevation of the sound source.

Combined with ITD/IPD and ILD this gives us a lot of information to work with to localise sound. Add in the fact that we can move our heads to get even more information and we can get precise localisation. In the next section we'll look at how Ambisonics can start to recreate accurate binaural hearing cues.

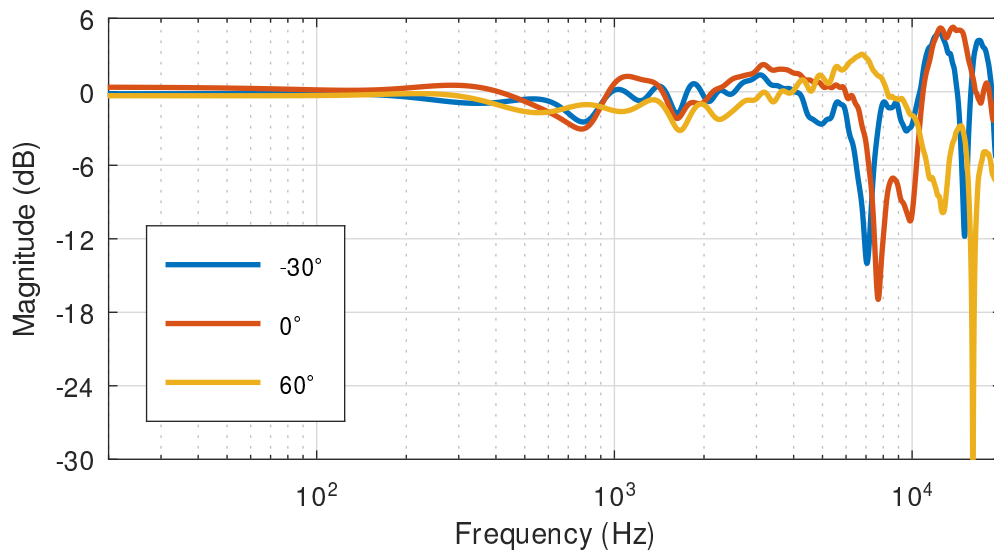


Figure 3.3: The spectra for the left ear of a Neumann KU100 dummyhead HRTF [14] for three different source elevation angles (-30° , 0° , 60°) with an azimuth of 0° .

Summary

The human auditory system uses various information to localise where a sound source is coming from: ITD, ILD, spectral cues and dynamic cues. These cues are encoded in the HRTF, which is unique to each listener due to different outer-ear, head and torso shapes.

3.2 Ambisonic-to-Binaural Decoding

Since one of the main ways we listen to Ambisonics is via binaural decoding we'll take a look at how well the important binaural hearing cues from section 3.1 are reproduced by a binaural decoder.

The real question is: how well does Ambisonics actually reproduce the cues we just described? And does higher order make a meaningful difference? The answer to both is yes and the figures below show exactly why.

We'll compare the binaural signal produced by an Ambisonic decoder with

the target binaural signal for the same sound-source direction. Then we'll look at the IPD and ILD reproduced and how they compare. We'll do all of this for different Ambisonic orders to see the benefit we get by adding all those additional channels.

Physical Reproduction

Let's start with a binaural decode of a source at different positions around the listener, encoded at first-, third- and tenth-order Ambisonics. We're interested in how well the Ambisonics decode is able to reproduce the target HRTF for all of the positions.

Figure 3.4 shows the absolute difference between decode and target averaged across all the positions in the HRTF. The results are shown as a function of the frequency along the x-axis. As the order increases the error stays low for more and more of the frequency range, representing more ideal decoding that matches the target. For the first-order (1OA) decoder the error starts to rise even below 1000 Hz. Third-order (3OA) gets us above 1000 Hz and you can usually hear a huge difference by going from 1OA to 3OA. For tenth-order (10OA), the error doesn't rise significantly until nearly 4000 Hz.

Any frequency ranges with low error compared to the target are likely to produce only small or imperceptible differences. We can see from the figures that more and more of the frequency range becomes holophonic as the order rises.

Low-Frequency Psychoacoustic Reconstruction

Now let's look at the key low-frequency localisation cue: ITD/IPD. As mentioned in section 3.1, this is an important cue up to around 1.5 kHz. We'll

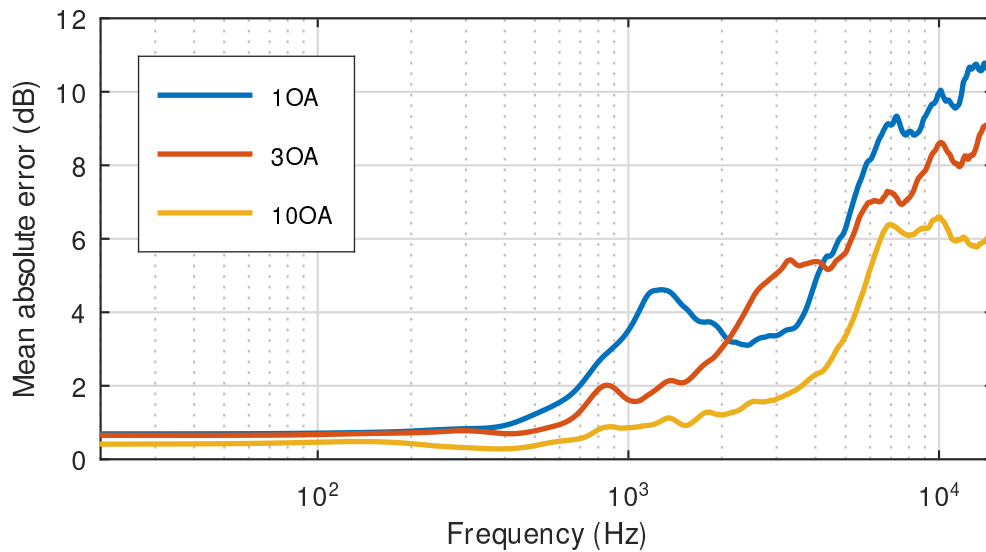


Figure 3.4: The mean absolute error for Ambisonics-to-binaural decodes of first-order, third-order and tenth-order. The error is low in the low frequency regions, with larger low-error ranges as Ambisonic order increases. Averaged over 20 subjects in the SADIE II database [14].

zoom in on that frequency range. Figure 3.5 shows the target ITD from the HRTF along with the difference between the target and the Ambisonically reproduced ITD for 1OA, 3OA and 10OA.

At low frequencies the 1OA signal is following the expected trend. But even at the quite low frequency of 400 Hz the ITD starts to diverge from the target and the error increases. Considering that ITD is relevant up to 1.5 kHz, this explains why localisation with 1OA can be blurry or unstable. Essentially, as the frequency increases the maximum ITD decreases away from the target, meaning the ITD cues are not consistent across frequency with what the auditory system expects.

For the higher orders this range is significantly increased. For third-order the ITD is already well reproduced across most of the relevant frequency range. This might be one reason that explains the large jump in spatial quality between first and third orders: The ITD cues are coherent across the required range.

For tenth-order the improvements are marginal because the benefits for

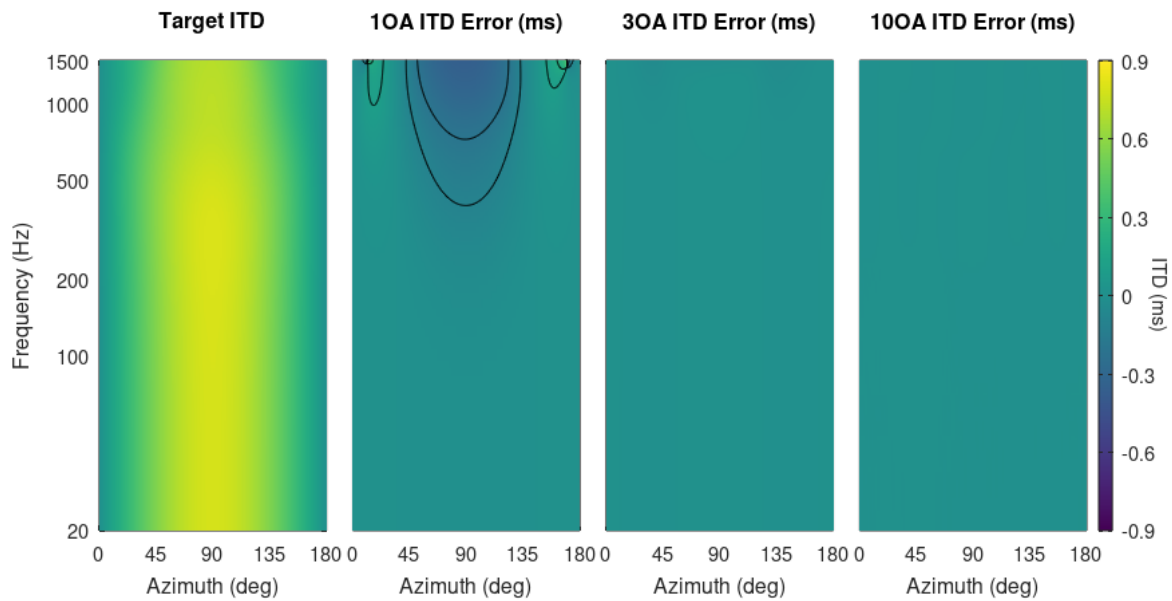


Figure 3.5: (Left) The target ITD for a real sound source as it moves around the listener. The remaining figures show the difference between the ITD recreated by 10A, 30A and 100A for the same positions. 10A shows that the error starts to increase with frequency, 30A shows only small error in higher frequencies, and 100A shows almost no error across the whole range. Note that the frequency range is limited to a maximum of 1.5 kHz.

these orders are in the higher frequencies. Things are already working well by third-order.

High-Frequency Psychoacoustic Reconstruction

The ILD is a dominant cue in the higher frequency ranges. As before, we'll look at the ILD for 10A, 30A and 100A and compare it to the target HRTF. It should be pretty clear at this point that increasing orders lead to better reconstruction of our key cues. And that is indeed what fig. 3.6 shows.

Note that for this decoder the $\max r_E$ optimisation (see section 2.6) has been used for higher frequency ranges above the point where ITD stops being well reproduced. The aim of this decoder optimisation is to concentrate the acoustic energy near the source direction, in order to help recreate correct ILD cues.

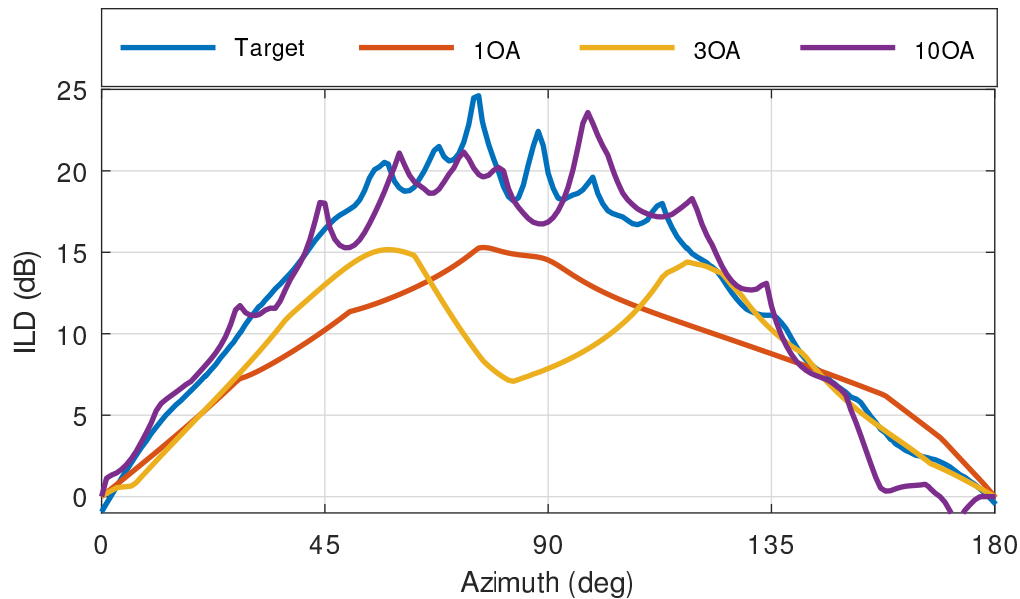


Figure 3.6: The average ILD for a real HRTF (target) and three orders of Ambisonics-to-binaural decoders. As the order increases, the ILD gets closer to the target.

At first order the ILD is low when compared to the target. The third-order decoder shows improvements for some positions but degrades again for sources from the sides. By tenth-order the ILD reproduction is not yet perfect but gives a good approximation of the target ILD.

The important thing to remember here is that for third-order and higher the ITD cues are well recreated, which will already help to provide a strong sense of localisation of the image. The ILDs are all created with broadly the correct trends but improvements with increasing order result in even more coherence with low-frequency ITD cues, leading to even more precise localisation. The decodes measured throughout this section were generated with aXMonitor at each order.

Exercise

Try filtering some white noise with low- and high-passes with different cutoffs. Adjust the order of the encoding on aXPanner. Compare when

you hear a difference between the orders.

Summary

Ambisonics can be decoded to binaural for high-quality spatial audio rendering. The higher the Ambisonic order, the larger the frequency range that is recreated with low error.

4 Tips for Working with Ambisonics

Goals

By the end of this chapter you will know:

- The conventions to check before you start, to avoid problems down the line.
- Which effects are safe to apply to an Ambisonic signal.
- How to check that your mix translates across different decoders, orders and playback formats.
- Practical and creative considerations when spatially composing a scene.

Working with Ambisonics is almost as straightforward as working with stereo, once you know the conventions, but there are some specific pitfalls that catch people out. There are also some techniques that are unique to spatial audio that can benefit your workflow.

Some of the tips in this chapter are mentioned earlier in the eBook. They're repeated here to have an easier reference later on.

4.1 Setup and Signal Integrity

Channel sorting matters

Just like 5.1, if you don't have the routing correct then spatial perception will be wrong. If the channels are in the wrong order, at best, sources will appear in

completely wrong positions. At worst, the signal's spatial image may collapse entirely into a diffuse, unlocalisable cloud.

Also like 5.1, there are different ordering conventions, and they're not compatible with one another. Thankfully, most modern plugins and tools use the *AmbiX* convention so it has become less and less of a problem over the years. But if you're using samples you find online you should be sure that the channel ordering is the one expected by your signal chain. The most common legacy ordering is Furse-Malham (FuMa), so that's one to watch out for in particular, because some Ambisonic microphone converters still support it.

Channel gains matter

Ambisonics uses the gain differences between channels to encode the spatial information. It also has different conventions for how it calculates these gains. The main conventions are SN3D (used by the *AmbiX* convention), N3D and FuMa. A gain convention mismatch typically produces an unwanted change in source width. Thankfully, as with channel sorting, this is handled automatically if all your tools support the *AmbiX* convention, which is almost always the case.

Along with gain convention mismatch, if gains are applied independently to each channel - for example, a multi-mono compressor - the spatial image will be altered (either position changes, width changes or complete collapse).

All this to say: be careful with any processing that changes Ambisonic channels independently. Inter-channel gain differences are the main way Ambisonics encodes directional information in the audio signal.

Not all effects are Ambisonic-safe

If an effect applies exactly the same processing to all channels then it's safe to use with an Ambisonic signal. However, if it introduces independent level, phase or time differences between channels then it can alter or damage the spatial information in the sound scene. Even a small unwanted phase difference can take your signal from an immersive wonderland to a diffuse, blurry mess. Ambisonics-specific processors are different: they may intentionally transform or mix the channels in a mathematically controlled way. All of the [aX Plugins](#) are designed to work with Ambisonics and preserve the spatial information. Most EQs should be fine as long as they process all channels identically and don't use independent per-channel dynamic processing. If you're not sure how an effect works, it's better to apply it to the source before converting to Ambisonics. If you are working in a DAW that supports multi-mono effects, like Pro Tools, you should be very wary of how they are processing each channel.

Warning!

Per-channel compressors, limiters, and denoising tools are among the most reliable ways to accidentally destroy a spatial image. A compressor that applies independent gain reduction to each channel will alter the inter-channel gain relationships that encode the sound field. Always apply dynamic range processing before encoding to Ambisonics where possible, or use tools specifically designed for Ambisonic signals, such as [aXCompressor](#), which is engineered to preserve the spatial relationships between channels.

Make sure your decoding order matches your signal

Most Ambisonic decoders will use different processing methods depending on the order of the signal they're provided, to optimise the decoding. If your decoder gives you the option to select the input order then you should be sure it matches the signal. It might be tempting to think that using a third-order decoder on a first-order signal will give you better results. In fact, the opposite is true! At best, you'll get a result that's identical to first-order decoding but lose some CPU. At worst, the decoder will apply optimisation specific to third-order and this will actually degrade the spatial quality of the final output.

Tip

aXMonitor (and all of the aX processing plugins) displays the order of the incoming signal on the bottom left of the UI. This gives you an easy way to verify what order it is decoding to, ensuring it matches your intention. It also adapts its internal processing to match the incoming order without action from you.

Warning!

Never assume your decoder is automatically detecting the order of your signal. Always verify it. A mismatch between signal order and decoder order is a common cause of unexpectedly poor spatial quality. It can be difficult to diagnose by ear alone because the result is a degradation of the spatial and timbral quality rather than obvious failure.

4.2 Monitoring and Mix-Checking

Use a head-tracker

Head-tracking lets you dynamically explore your sound field. This is particularly great if you're targeting VR/AR. Supperware¹ makes a head-tracker that is compatible with the aXRotate plugin, as well as many other plugins, so you can use it while mixing right in your DAW.

Listen to different decoding options

Unless you know your mix is only going to be heard via a single decoded version, you should check your mix with different decoders. aXMonitor lets you compare stereo and binaural versions. If you're targeting loudspeaker playback then you can try different decoders on the same layout. Think of it like checking your mix in the car.

Check the mix for different orders

You're mixing a seventh-order project that will have different distribution formats. Some are using the full seventh-order. Some truncate to third-order. One even only decodes first-order! The advantage of Ambisonics is that you only need to mix once at seventh-order. But you should definitely check your mix at different orders to ensure it translates well. Most of the time it will be fine but maybe you'll lose some detail at lower orders and that will make you rethink some creative choices to get the best result across orders.

¹<https://supperware.co.uk/headtracker-overview>

Try different HRTFs

Depending on your distribution target you have different choices for how to work with HRTFs. If you are going to be distributing a pre-decoded version (let's say you have to provide a 2-channel mix for Spotify) then you should pick a good HRTF and stick to mixing through that. Easy. Now, let's say you're targeting a streaming platform that lets you upload the raw Ambisonic file and it handles the decoding. In that case you'd want to try to find their reference HRTF or decoder. This isn't always possible but aXMonitor does include the HRTF used by Google for YouTube. Finally, maybe you're mixing binaurally but targeting different distribution formats (streaming, live diffusion at a concert, VR/AR). In that case you'd probably want to either get your own HRTF measured (the best solution but not easy!) or test different ones from different databases until you find one that works well for you.

Check the stereo fold-down

A significant proportion of listeners might hear your mix as a stereo decode, whether because they are listening on a standard speaker system, using a platform that handles decoding automatically, or simply playing back on a device that does not support spatial audio. Use aXMonitor's stereo decode option to check how your mix translates. The spatial character will be reduced, but the essential elements should remain identifiable and the overall balance should hold. If something sounds dramatically wrong in the stereo version then consider adjusting the panning or level of that element in the Ambisonic scene.

4.3 Mixing Considerations

Think space, not panning

In stereo you balance elements along a line between two speakers. In Ambisonics you are placing them on a sphere around the listener, so every panning decision now has an azimuth and an elevation to consider. As with stereo, two sources that mask each other when they share a position can often be separated by moving one of them rather than by reaching for an EQ. Elevation is an easy dimension to neglect: then everything ends up sitting on the horizontal plane, which throws away a good part of what the format offers. It also helps to think about the scene as a whole rather than as a stack of tracks.

Consider processing before panning

For mono and stereo sources that you are panning into the Ambisonic scene, applying EQ and dynamic processing *before* encoding is both more efficient and lower-risk. Unless you are applying the processing to multiple tracks, in which case it could make sense to apply it as a bus-effect post-panning. Because EQ and Ambisonic panning are linear operations, the order does not affect the result. For Ambisonic microphone recordings, where sources cannot be separated, use Ambisonics-compatible processors on the full signal.

Mic placement depends on context

Ambisonic microphones are great at capturing sources along with the acoustic space. Where you place the microphone will control how much direct sound there is compared to the reverberation (direct-to-reverberant energy ratio or D/R). The principle is no different to traditional mono or stereo re-

cording. Placing the microphone close to the source gives a high D/R. Placing it far away gives a low D/R. It's possible to record a whole scene with only a single Ambisonic microphone. With appropriate placement virtual microphones (beamforming) could be used to follow the sources and even remix to other formats later. In fact, if done well, you could record a whole performance with a single Ambisonic microphone and just rely on a good decoder. Imagine recording a string quartet or a choir with one microphone. Another option would be to place the microphone further back and to close mic the main sources. Then the close mic recordings can be panned to Ambisonics and the Ambisonic microphone can be used to provide the ambient bed.

4.4 Creative Considerations

Higher order isn't always better

For panned sources, working at a higher order can preserve more spatial resolution and can be useful for future-proofing. However, you have to balance that against CPU load, storage and your actual delivery requirements. While I think it's always best to mix in the highest order possible (tenth-order with the [aX Plugins](#)) for future-proofing, you have to balance the CPU load and storage increase with your practical requirements. If you're working on something that will be played on limited hardware then maybe you're stuck with first-order, so work with that. Or maybe you're constantly working on Ambisonic projects that target third-order. Sure, it would be nice to keep tenth-order for later but then you'd have to fill your studio with external drives to store them, all for a just-in-case scenario. How you balance it is up to you.

Another consideration, if you're working with panned signals, is that higher

orders lead to more precise localisation. This is usually a good thing but there could be times when it feels too much like a “spotlight” on the sound. Trying a lower order could lead to a scene that “blends” together more naturally. This will be very dependent on your signals and, perhaps, artistic choices.

Don't underestimate psychoacoustics

Our brain is amazingly powerful. It can use context clues to override the actual physical cues that we hear if they don't match. The classic example of this is the *ventriloquist effect*, where a sound appears to be coming from a different visible source than the true direction. The visual cue effectively overrides the audio one [16].

A purely audio example is taking a signal like a helicopter fly-over or a bird chirping. A lot of the time signals like this will sound “up” no matter where they're placed, as long as the sound isn't coming from a single loudspeaker². Basically, we're just used to sounds like these coming from a certain direction so our brain overrides acoustic cues.

When you're crafting your scenes you should be careful that there is some coherence between the signal and its “expected” position. And, of course, you can use this to your advantage! Make up for a lack of elevated speakers by using the listener's brain to “fill in the gaps”.

Tip

When placing sounds in your scene, consider what position the listener will *expect* them to come from based on the signal content alone. A thunderclap placed below the horizon will often still be perceived as coming

²I once was part of an informal experiment in a concert space with speakers *below* the audience. We tested playing some audio of birds and asking people where they were. People pointed above them even though the audio was panned below their feet! It's a powerful effect.

from above. You can use this to your advantage.

Let your ears guide you

Maybe you did something that's technically "wrong" but it sounds good to you. Ambisonics gives you a genuinely new set of creative tools (spatial delay, sound field rotation, direction dependent processing) and the best way to learn what works is to experiment as much as possible. The technical knowledge in this eBook gives you a foundation, but your ears and your artistic instincts should have the final say.

Summary

Like any audio format, Ambisonics has some specific quirks to be aware of. The ordering, gain and phase relationships between channels are vital to conserve. Checking your mix using different decoding methods or formats will help you get the best results that translate to different situations. The most important rules are to make sure the intended relationships between channels don't change, to keep Ambisonic channel ordering and normalisation consistent (usually AmbiX), and to make sure the decoder matches the signal order.

What Next?

You now have a solid foundation in Ambisonics: how it works, why it sounds the way it does, and how to use it in your projects. The best thing you can do from here is start experimenting.

If you haven't already, download a free 15-day trial of the [aX Plugins](https://www.ssa-plugins.com/downloads/) from www.ssa-plugins.com/downloads/ and work through the exercises at your own pace.

Questions, feedback, or just want to share something you've made? You can reach me directly at peter@ssa-plugins.com.

Ambisonics Quick Reference

Record/Pan → Process → Decode → Listen

Order

10A = 4 ch	20A = 9 ch
30A = 16 ch	40A = 25 ch
70A = 64 ch	100A = 121 ch

Higher order means greater **spatial resolution**, but more CPU, storage and channels.

Use an order appropriate to the source and delivery format. Consider mixing to the highest order possible to keep a future-proof master.

Higher-order processing cannot add spatial information that was not present in the source.

Format

For most modern workflows:

$$\text{AmbiX} = \text{ACN} + \text{SN3D}$$

Watch for legacy **FuMa** material.

If positions or source width seem wrong, check the **channel ordering and normalisation**.

Processing

Ordinary effects: keep processing linked across all Ambisonic channels.

Avoid independent per-channel gain, compression, limiting, delay or phase changes.

For mono/stereo sources, apply ordinary EQ and dynamics **before panning** when practical.

Ambisonic processors may deliberately transform the channels while preserving the sound field.

Recording

Ambisonic mic → ambience + realism

Panned source → precision + control

Combining the two can give the best of both.

Keep the original **A-format/raw recordings**.

Monitoring

- Match the **decoder order** to the signal.
- Keep the full channel count through every track and bus.
- Check lower orders when they will also be delivered.
- Check stereo when listeners may receive stereo.
- Try different HRTFs when binaural translation matters.

For mixed-order projects, decode each order separately before summing when the workflow allows it.

Before Delivery

- Check **order + channel count**.
- Check **ACN/SN3D**.
- Check the **decoder**.
- Check track, bus and master **channel widths**.
- Check current **platform requirements**.

Keep an **Ambisonic master** as well as the final renders.

Something Sounds Wrong?

Wrong direction → channel ordering

Too wide/diffuse → normalisation or decoder order

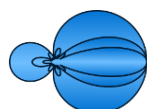
Missing detail → channel truncation

Changes after an effect → independent channel processing

Poor binaural localisation → order/HRTF

When in doubt

Format → Channel Count → Decoder



SSA
P L U G I N S

ssa-plugins.com

▪ *Getting to Know Ambisonics*

About the Author

I have been working with Ambisonics since 2011 when I completed my Master's degree at the exceptional Sonic Arts Research Centre (SARC) at Queen's University Belfast. My first exposure to spatial audio in general was getting to hear some amazing compositions by world-class composers in SARC's Sonic Lab - a huge custom multichannel space with dozens of speakers, even under your feet! During the course, we were allowed to pick a project and I decided to make VST plugins for encoding and decoding for playback in the Sonic Lab. I was pretty much hooked from that point onward. During my time there and in the years since then I've worked with artists to do the spatialisation and sound design for public exhibitions in museums, public events and even in an outdoor garden at a French chateau. For a while I was a researcher in France, where I got to learn a lot more about psychoacoustics and dynamic, motion-tracked systems. In fact, I occasionally still take part in research-based projects because I think it's important and, more importantly, it's fun! I released the aX Plugins under the SSA Plugins banner in 2018 to help make Ambisonics even more accessible to audio pros used to working with stereo or traditional surround. Everything in this eBook comes from that experience and from wanting to make Ambisonics as accessible as it deserves to be.

Appendix A DAW Setup and Example Projects

Goals

In this appendix you will download DAW projects and install plugins to follow along with the exercises throughout this eBook.

An easy way to learn about Ambisonics is to be able to hear how the theory is reflected in the audio. To do this, I've made a set of projects for Pro Tools, Reaper, and Nuendo/Cubase you can download from www.ssa-plugins.com/ebook-daw-projects. Any time you see an **Exercise** box you can go to the referenced project and experiment for yourself.

To follow along, you will also need to download and install the aXBundle (available from the same link). There is a 15-day free trial, so you can work along with these examples for free during that time. Some projects also include the free aXMeter plugin (available [here](#)).

Exercise

Once you've installed the plugins and downloaded the example projects, open the OrderComparison project to check everything is working. You should hear the sound moving around you in a circle, starting from the front and going clockwise (right, back, left, then back to the front).

Each project should be functionally equivalent between the DAWs, but there are some quite significant differences between how each DAW handles Ambisonics or multichannel tracks.

We'll explore some of the main points to be aware of here to help you set up your own projects. The maximum Ambisonic order that each DAW supports can vary widely. Table A.1 shows the maximum order supported by the DAWs used for the examples. The next sections include specific information about these DAWs, which you can always come back to if you need to troubleshoot your own project setup.

DAW	Max. Order
Reaper	10
Pro Tools Ultimate	7
Pro Tools Studio	3
Nuendo 15	4
Cubase 15	4

Table A.1: The maximum Ambisonic order supported by a selection of DAWs.

Tip

If your usual DAW isn't included here (or doesn't support Ambisonics at all) you can follow along in the examples using Reaper. They offer a free trial which should be long enough to work through the examples here.

Pro Tools

Pro Tools makes it very easy to work with Ambisonics. Tracks can either be natively Ambisonics (called 1OA, 2OA, 3OA and up to 7OA in Pro Tools Ultimate) or stereo tracks can be converted to Ambisonics by inserting an appropriate AAX plugin. If you have some Ambisonic audio you can insert it directly on the Ambisonic track of the appropriate order and just route it like you would any other track. Alternatively, you could add a mono or stereo track and then add an Ambisonic panner as an insert effect. Pro Tools will then treat all downstream audio as Ambisonics. Similarly, if a decoding plugin

is added then it will convert from Ambisonics to the specified output format, which is then routed as normal.

Overall, working with Ambisonics in Pro Tools should be intuitive for anyone with experience working with different track types.

Reaper

Reaper does not have a concept of track types in the same way as Pro Tools. Instead, the total number of channels can be set on a per-track or bus basis. In order to ensure that a track has enough channels to support Ambisonics you should set the **Track channels** to at least the number required to support the Ambisonic order you're working with. This can be done by clicking on the **Route** button on the track and adjusting the **Track channels** setting. It can also be done by clicking on the routing button above the top right of the Ambisonic plugin GUI after adding it. By default, this will usually start out as 2in/2out. You can enter the number of channels required on the bottom left of the window that pops up.

Warning!

Reaper does not propagate channel counts downstream. If you set a track to 16 channels (third-order) and route it to the Master bus, the Master bus will still default to 2 channels and silently discard everything above stereo. You must manually set the correct channel count on **every** track and bus in the signal chain, including the Master.

Nuendo and Cubase

Nuendo 15 and Cubase 15 both handle Ambisonics in the same way, supporting first- to fourth-order. Cubase/Nuendo allow you to select the track **Configuration**. If you select 1OA/2OA/3OA/4OA then you can either import Ambisonic native audio or insert a panner as an effect. Alternatively, you can set up a Group track, or a Folder track with Group Channel enabled, using an Ambisonic configuration, and route your mono or stereo tracks to it. Then, you can change the track panner to insert an Ambisonic panner by right-clicking on the panner UI element in the mixer.

Tip

Try setting up a project with Ambisonic tracks and buses in your DAW using the information in the appendix. To check if everything is correctly configured, add aXMeter just before the aXMonitor decoder. This will show you for sure that all of the intended channels are arriving at the decoder. If you see too few channels with signal present then go back and check the track widths along each point in the signal chain.

Once you have your project configured, save it as a template. Having a ready-made Ambisonic setup will save you significant time on future projects.

Summary

You now have the projects you need to follow along with the exercises in this eBook. Pro Tools, Reaper, and Nuendo/Cubase all handle Ambisonics slightly differently, but the core concepts are the same across all of them. Also, consider saving your own project template once you're set up.

Appendix B Glossary

10A, 20A, etc.

Denotes an Ambisonic signal of the specified order i.e. 10A = first-order Ambisonics, 30A = third-order Ambisonics.

A-format The raw microphone signals recorded by an Ambisonic microphone. These signals should not be listened to directly. Instead, they should be converted to B-format, which is the encoded Ambisonic signal. This conversion takes place using (ideally) a manufacturer-supplied converter.

ACN *Ambisonic Channel Number.* The standard channel ordering convention used in the AmbiX format. Channels are numbered sequentially from zero, beginning with the omnidirectional component (W) and proceeding through each order in a defined pattern. ACN is part of the AmbiX convention and has become the de facto standard for modern Ambisonics workflows.

AIIRAD *All-Round Ambisonic Decoder.* A decoding method that works well for irregular loudspeaker layouts such as 5.1.4. It decodes the Ambisonic signal to an ideal set of virtual loudspeakers distributed evenly across the sphere, then uses amplitude panning to map each virtual speaker to the real loudspeaker layout. A reliable default choice for most practical speaker configurations.

AmbiX The current standard convention for encoding Ambisonics signals, combining ACN channel ordering with SN3D normalisation. Ad-

opted by most modern plugins, DAWs, and streaming platforms. Unless you are working with legacy material, AmbiX is almost certainly the convention your tools are using.

Ambisonic Order

A measure of the spatial resolution of an Ambisonic signal. Higher orders encode more spatial detail and require more channels: first-order uses 4 channels, second-order uses 9, third-order uses 16, and so on, following the formula $(N + 1)^2$ for order N . Third-order is generally considered a practical sweet spot between quality and channel count, though higher orders provide more spatial fidelity.

B-Format The encoded Ambisonic signal itself, comprising the full set of spherical harmonic channels for a given order. A first-order B-Format signal contains four channels: W (omnidirectional), X (front-back), Y (left-right), and Z (up-down). Higher-order B-Format signals contain additional channels that encode progressively finer spatial detail.

Binaural Audio

A two-channel audio format designed for headphone playback that uses Head-Related Transfer Functions (HRTFs) to simulate sounds arriving from specific directions in three-dimensional space. Ambisonics can be decoded to binaural, making it one of the most common ways to monitor and distribute spatial audio mixes.

Cone of Confusion

A cone-shaped surface centred on the listener's head along which all points produce identical ITD and ILD cues. Because sounds

anywhere on this cone reach the ears with the same timing and level differences, the auditory system cannot distinguish between them using ITD and ILD alone. Additional cues such as spectral information and head movement are needed to resolve the ambiguity.

Duplex Theory

A model of sound localisation describing how the auditory system uses different cues at different frequencies. Interaural Time Difference (ITD) dominates localisation below approximately 1.5 kHz, while Interaural Level Difference (ILD) takes over at higher frequencies. Spatial audio systems must reproduce both cues accurately to achieve stable, convincing localisation.

EPAD *Energy Preserving Ambisonic Decoder.* A decoding method that aims to maintain consistent loudness regardless of source position, avoiding the level fluctuations that can occur with irregular loudspeaker layouts. A useful alternative to AllRAD when loudness consistency across the listening space is more important than maximum localisation precision.

Externalisation

The perceptual quality of a binaural sound source appearing to originate outside the listener's head, rather than inside it. Poor externalisation causes sounds to feel “in-head” rather than spatially placed in the environment. Higher Ambisonic orders and personalised HRTFs both improve externalisation.

FOA/1OA

First-Order Ambisonics. The simplest form of Ambisonics, using four channels to encode a full 360° spherical sound field. Conceptually an extension of the stereo mid-side technique to three dimensions. 1OA is the most widely supported order and is commonly produced from tetrahedral microphone arrays after A-to-B conversions. FOA is a common abbreviation.

Front-Back Confusion

A common localisation error in binaural audio in which the auditory system perceives a sound positioned in front as coming from behind, or vice versa. It occurs because sounds at certain front and rear positions can produce similar ITD and ILD cues. Small head movements and accurate spectral cues in the HRTF both help to reduce front-back confusions.

FuMa

Furse-Malham. A legacy Ambisonic channel ordering and normalisation convention, common in older software and sample libraries. Not compatible with the AmbiX convention. The most notable difference in first-order is that FuMa scales the W channel down by approximately 3 dB. If you encounter Ambisonic files that sound spatially incorrect, a FuMa-to-AmbiX conversion is often the cause.

Head Tracking

A system that detects the orientation of the listener's head in real time and applies a corresponding counter-rotation to the Ambisonic sound field, so that sound sources remain perceptually fixed in space as the head moves. Head tracking significantly

improves externalisation and front-back localisation, and is particularly valuable in VR and AR applications.

HOA *Higher-Order Ambisonics.* Any Ambisonic signal beyond first-order. HOA signals encode greater spatial resolution at the cost of a higher channel count. Second-order (9 channels) and third-order (16 channels) are the most common HOA formats in practical use.

Holophonic

Describing the property of an Ambisonic decode that physically recreates the intended sound field at the listening position, analogous to a hologram for audio. In practice, the holophonic region becomes smaller as frequency increases, and higher Ambisonic orders extend the frequency range over which accurate physical reproduction is achieved.

HRTF *Head-Related Transfer Function.* A set of direction-dependent filters that describe how sound is modified by the shape of a listener's head, ears, and torso before reaching the eardrums. HRTFs encode ITD, ILD, and spectral cues for every direction around the listener. Because each person's anatomy is unique, HRTFs are individual: using someone else's HRTF can degrade localisation quality and timbral accuracy.

ILD *Interaural Level Difference.* The difference in loudness between the two ears caused by the acoustic shadow of the head. ILD is the primary localisation cue at frequencies above approximately 1.5 kHz. Higher Ambisonic orders reproduce ILD more accurately, particularly for sources positioned to the sides of the listener.

In-Phase Decoding

A decoder optimisation that ensures loudspeakers on the opposite side of the array from the intended source direction are inactive. This reduces the precision of localisation at the sweet spot but extends the usable listening area, making it a good choice for large audiences or diffuse playback environments.

IPD *Interaural Phase Difference.* The phase offset between the signals arriving at the two ears, from which Interaural Time Difference can be derived. IPD is used by the auditory system for localisation at low frequencies, up to the point where the wavelength becomes shorter than the distance between the ears and phase ambiguity makes it unreliable.

ITD *Interaural Time Difference.* The difference in arrival time of a sound between the two ears. ITD is the dominant localisation cue below approximately 1.5 kHz. First-order Ambisonics reproduces ITD accurately only at low frequencies; higher orders extend accurate ITD reproduction further up the frequency range.

MAD *Mode-Matching Ambisonic Decoder.* A decoding method that requires a mathematically well-conditioned loudspeaker layout that provides good coverage of the sphere. Rarely appropriate for practical, irregular layouts such as 5.1 or 7.1.4, but included here for completeness.

Max r_E Decoding

A psychoacoustically motivated decoder optimisation that concentrates loudspeaker energy in the direction of the intended source,

rather than allowing all speakers to be active simultaneously. Used in the high-frequency band of most advanced decoders to improve ILD reproduction. Complements basic decoding, which is used in the low-frequency band where full speaker interference is needed to recreate the sound field physically.

N3D *Full three-dimensional normalisation.* A normalisation convention for Ambisonic channels that scales each channel so that the mean-squared value of its directivity pattern is unity. Used in some tools and formats. Not compatible with SN3D without a gain correction; a mismatch between the two will alter source width and spatial imaging.

Normalisation

In the context of Ambisonics, the convention used to set the relative gain of each channel in the B-Format signal. The two most common conventions are SN3D (used in AmbiX) and N3D. A normalisation mismatch between encoder and decoder typically manifests as an unexpected change in source width or, in severe cases, a complete collapse of the spatial image.

SAD *Sampling Ambisonic Decoder.* The simplest decoding method, which points a directional virtual microphone toward each loudspeaker and samples the sound field in that direction. Produces acceptable results only for perfectly regular loudspeaker layouts (e.g. a sphere of evenly spaced speakers). Most applicable to binaural decoding. Not recommended for practical irregular layouts.

SN3D *Semi-normalised three-dimensional normalisation.* The normalisa-

tion convention used in the AmbiX standard. In SN3D the gain of each channel is bounded so that no channel exceeds unity for any direction. The most widely used convention in modern Ambisonics tools and the one you will encounter in almost all current plugins and DAWs.

SOFA *Spatially Oriented Format for Acoustics.* A standardised file format for storing HRTFs and other spatial audio measurements, enabling easy exchange between tools and platforms. Several freely available SOFA databases exist online, allowing you to audition different HRTFs and find one that provides accurate spatial impression for your own ears.

Spherical Harmonics

The mathematical basis functions used to decompose and represent a sound field in Ambisonics. Each Ambisonic channel corresponds to one spherical harmonic, with increasingly complex directivity patterns as the order increases. First-order spherical harmonics correspond to the familiar omnidirectional and figure-8 polar patterns; higher-order harmonics produce the more intricate petal-like patterns seen in HOA signals.

Sweet Spot

The listening position at which an Ambisonic decode most accurately recreates the intended sound field. The size of the sweet spot decreases as frequency increases, and grows with higher Ambisonic orders. In-phase decoding can be used to trade some localisation precision for a larger usable listening area.

W Channel

The omnidirectional component of a B-Format signal, capturing the overall pressure of the sound field equally from all directions. In first-order Ambisonics it corresponds to the “Mid” channel in a mid-side analogy. Present in all Ambisonic orders and always the first channel in both the FuMa and AmbiX conventions.

Yaw, Pitch, Roll

The three axes of rotation applied to an Ambisonic sound field during rotation processing or head tracking. Yaw is rotation around the vertical (Z) axis; pitch is rotation around the left-right (Y) axis; roll is rotation around the front-back (X) axis. The order in which rotations are applied matters mathematically; the yaw-pitch-roll convention is most commonly used.

Bibliography

- [1] Michael Gerzon. Periphony: With-Height Sound Reproduction. *Journal of the Audio Engineering Society*, 21(1):2–10, 1973.
- [2] Alan Blumlein. Improvements in and relating to sound-transmission, sound-recording and sound-reproducing systems. *British patent no. 394325*, 1931.
- [3] Jérôme Daniel. *Représentation de champs acoustiques, application à la transmission et à la reproduction de scènes sonores complexes dans un contexte multimédia*. PhD thesis, University of Paris 6, 2000.
- [4] John F. Culling, Sophia C. MacLeod, and Joseph A. Shiebert. Effect of ambisonic order on sound localization in the horizontal plane. *The Journal of the Acoustical Society of America*, 160(1):22–33, July 2026. doi: 10.1121/10.0044227.
- [5] Boaz Rafaely. Analysis and design of spherical microphone arrays. *IEEE Transactions on Speech and Audio Processing*, 13(1):135–143, January 2005. doi: 10.1109/TSA.2004.839244.
- [6] Durand R. Begault, Elizabeth M. Wenzel, and Mark R. Anderson. Direct comparison of the impact of head tracking, reverberation, and individualized head-related transfer functions on the spatial perception of a virtual speech source. *Journal of the Audio Engineering Society. Audio Engineering Society*, 49(10):904–916, 2001.

- [7] W. Owen Brimijoin, Alan W. Boyd, and Michael A. Akeroyd. The contribution of head movement to the externalization and internalization of sounds. *PLoS ONE*, 8(12):1–12, 2013. ISSN 19326203. doi: 10.1371/journal.pone.0083068.
- [8] Christof Faller, Fritz Menzer, and Christophe Tournery. Binaural audio with relative and pseudo head tracking. In *138th Convention of the Audio Engineering Society*, Warsaw, Poland, 2015.
- [9] A. W. Mills. On the minimum audible angle. *The Journal of the Acoustical Society of America*, 30(4):237–246, 1958.
- [10] MA Poletti. Three-Dimensional Surround Sound Systems Based on Spherical Harmonics. *J. Audio Eng. Soc*, 53(11):1004 –1025, 2005.
- [11] F L Wightman and D J Kistler. Resolution of front-back ambiguity in spatial hearing by listener and source movement. *The Journal of the Acoustical Society of America*, 105(5):2841–53, May 1999.
- [12] Elizabeth M. Wenzel, Marianne Arruda, Doris J. Kistler, and Frederic L. Wightman. Localization using nonindividualized head-related transfer functions. *The Journal of the Acoustical Society of America*, 94(1): 111–123, 1993.
- [13] Piotr Majdak, Franz Zotter, Fabian Brinkmann, Julien De Muyne, Michael Mihocic, and Markus Noisternig. Spatially Oriented Format for Acoustics 2.1: Introduction and Recent Advances. *Journal of the Audio Engineering Society*, 70(7/8):565–584, July 2022. doi: 10.17743/jaes.2022.0026.

- [14] Cal Armstrong, Lewis Thresh, Damian Murphy, and Gavin Kearney. A Perceptual Evaluation of Individual and Non-Individual HRTFs: A Case Study of the SADIE II Database. *Applied Sciences*, 8(11):2029, November 2018. doi: 10.3390/app8112029.
- [15] John W. Strutt. On our perception of sound direction. *Philosophical Magazine*, 13:214–232, 1907.
- [16] David Alais and David Burr. The Ventriloquist Effect Results from Near-Optimal Bimodal Integration. *Current Biology*, 14(3):257–262, February 2004. doi: 10.1016/j.cub.2004.01.029.