Optimized Decoders for Mixed-Order Ambisonics*

Aaron Heller[1], Eric Benjamin[2], and Fernando Lopez-Lezcano[3]

[1]*Artificial Intelligence Center, SRI International, Menlo Park, CA*
[2]*Surround Research, Pacifica, CA*
[3]*Center for Computer Research in Music and Acoustics (CCRMA), Stanford University, Stanford, CA*

May 23, 2021

Abstract

In this paper we discuss the motivation, design, and analysis of ambisonic decoders for systems where the vertical order is less than the horizontal order, known as mixed-order Ambisonic systems. This can be due to the use of microphone arrays that emphasize horizontal spatial resolution or speaker arrays that provide sparser coverage vertically. First, we review Ambisonic reproduction criteria, as defined by Gerzon, and summarize recent results on the relative perceptual importance of the various criteria. Then we show that using full-order decoders with mixed-order program material results in poorer performance than with a properly designed mixed-order decoder. We then introduce a new implementation of a decoder optimizer that draws upon techniques from machine learning for quick and robust convergence, discuss the construction of the objective function, and apply it to the problem of designing two-band decoders for mixed-order signal sets and non-uniform loudspeaker layouts. Results of informal listening tests are summarized and future directions discussed.

1 Introduction

There is a renewed interest in decoders for mixed-order Ambisonics due to the availability of mixed-order microphones and the current COVID-19 restrictions placing an emphasis on loudspeaker arrays that can be deployed in domestic settings, where it is relatively easy to deploy a third-order horizontal array comprising eight loudspeakers. However, installing more than a few elevated speakers is difficult and placing speakers significantly below the listener is nearly impossible. While mixed-order operation is frequently cited as an advantage of Ambisonics, little has been written about creating or analyzing the performance of decoders specifically for mixed-order signal sets or highly non-uniform loudspeaker arrays. We also introduce a new implementation of the Ambisonic Decoder Toolbox (ADT) in Python/NumPy, which includes a fast and robust non-linear optimizer and a new design procedure for dual-band decoders where we first optimize the high-frequency performance of the decoder and then optimize the low-frequency performance to match the high-frequency [1].

2 Ambisonics

Ambisonics is an extensible, hierarchical system for representing sound fields. It defines how something should sound as opposed to specifying the signals going to particular speakers. Sound fields can be recorded using an Ambisonic microphone or created using an Ambisonic panner to position a sound in full 3D space. It is an isotropic representation of the sound field that can be rotated in the renderer making it attractive for virtual and augmented reality applications.

An Ambisonic signal set is a representation of the sound field as the time-varying coefficients of a spherical har-

*This paper has been accepted for presentation at the 150th AES Convention, May 25-28, 2021 (virtual).

[1]aaron.heller@sri.com
[2]ericmbenj@gmail.com
[3]nando@ccrma.stanford.edu

monic series. The spatial accuracy increases with the number of harmonics being used. A first-order Ambisonic signal set is four channels wide, third-order is sixteen channels, fifth order is 36, and so forth. Each increase in Ambisonic order adds spherical harmonics to the signal set and increases the spatial accuracy of the representation of the sound field. We use a shorthand notation to specify the signal set. For example 3H2V means third-order horizontal, second-order vertical, with the set of spherical harmonics according to the HV convention [2].

Once an Ambisonic signal set has been captured or generated, appropriate speaker feeds are produced by a decoder. Designing an optimal decoder, specifically the low- and high-frequency matrices, for a given signal set and loudspeaker array is the central topic of this paper. Other aspects of decoder design have been covered in earlier papers by the present authors [3].

2.1 Mixed-Order Ambisonics

A physical encoder (an Ambisonic microphone) needs to have enough capsules covering the sphere to accurately sample the spherical harmonics of the order it is intended to capture. Conversely, a speaker array needs to have enough loudspeakers covering the sphere to excite the spherical harmonics for the maximum order it is intended to reproduce. That is not always the case, leading to arrays with different densities of transducers in different directions. The consequence is that the order that can be encoded or decoded will change according to the direction.

For example, nine years ago, one of the present authors published the design for a second-order ambisonic microphone [4]. There have been four proprietary [5, 6, 7, 8] and one free and open-source implementation [9] of this design. A compromise made was to use only eight capsules. This simplifies calibration and allows the use of widely-available eight-channel recorders.

While commonly referred to as a second-order microphone, only eight of the nine spherical harmonic components needed for the second-order signal set can be derived from the capsule signals. The missing spherical harmonic is degree 2 and order 0, which is called "R" in the Furse-Malham convention. R is a "zonal" harmonic and varies only with elevation. Eliminating this component coarsens the description of the sound field at elevations other than horizontal, making it a 2HV1 mixed-order encoder. As we shall see, decoding this signal set with a decoder designed for full second order is suboptimal.

Small speaker arrays with a limited number of speakers in the vertical direction are another case in which the array does not have uniform density of speakers and cannot excite the spherical harmonics in all directions equally. Physical restrictions in the placement of speakers can also dictate that an array might not be capable of rendering the same order in both the horizontal and vertical directions. Such an array will need a mixed-order decoder.

3 Ambisonic Decoders

The task of the decoder is to create the best perceptual impression possible that the sound field is being reproduced accurately, given the available resources. In practical terms, the following criteria are necessary:

1. Constant amplitude gain for all source directions
2. Constant energy gain for all source directions
3. At low-frequencies, correct reproduced wavefront direction and velocity (Gerzon's velocity-model localization vector, $\mathbf{r_V}$)
4. At high-frequencies, maximum concentration of energy in the source direction (Gerzon's energy-model localization vector, $\mathbf{r_E}$)
5. Matching high- and low-frequency perceived directions ($\hat{\mathbf{r_E}} = \hat{\mathbf{r_V}}$)

Recent work shows that (4) is the most important [10]; it is also the most difficult to get right. After that, (2) and (5) are important, as it is thought that we use a majority voting system to resolve conflicting directional cues [11]. Decoders that ignore (5) can be fatiguing due to conflicting perceptual cues [12]. Note that to satisfy all of these criteria we must use decoders that have different gain matrices for high and low frequencies, so-called "two-band" or "Vienna" decoders [13].

The ADT includes a full-featured decoder engine written in the FAUST DSP specification language [14] that implements dual-band decoding, near-field correction, and level and time-of-arrival compensation. The ADT incorporates several design techniques that produce decoders that perform well according to these criteria for partial-coverage loudspeaker arrays, such as domes and stacked rings, but assumes that within those limits the speakers are (more or less) uniformly distributed. It also assumes that the decoders produced by these techniques are optimal for mixed-order signal sets.

3.1 Mixed-Order Decoders

Many diffusion systems simply use full-order decoders for mixed-order signal sets, leaving the missing channels unconnected, such as "R" in the case of the eight-capsule microphone described above. In Ambisonics, omitting channels from a signal set and leaving those channels unconnected and silent in a full-order signal set are two different things. In the former case, the decoder assumes a point source where the omitted components are not known. In the latter case, the decoder assumes that those components are known and that the spatial distribution of the sources is such that the silent components are exactly zero. The latter case would be extremely rare in real acoustic scenes.

This was investigated and we found that in every case examined, a decoder specifically designed for the mixed-order signal set outperformed a full-order decoder by the criteria listed at the beginning of this section.

As an example, Figure 1 shows the directional error of 3H1V and 2H1V signal sets being decoded by a 3H3V decoder (a and c) vs. a mixed-order decoder designed for the specific signal set (b and d). The speaker array is a small 8+5 array. A mixed-order decoder performs much better in terms of accuracy of the rendered directions.

4 Designing Decoders

For regular speaker arrays (2D polygons, 3D polyhedra, t-designs) the design of a correct decoder is a straightforward task:

- Build the speaker encoding matrix, $\mathbf{K}$, by sampling the spherical harmonics at the speaker directions.
- Use the pseudoinverse to find the basic decoding matrix, $\mathbf{M}$.
- Modify the per-order gains of $\mathbf{M}$ to maximize the magnitude of $\mathbf{r_E}$.

For this type of array, Gerzon proved that $\mathbf{r_E}$ will point in same direction as $\mathbf{r_V}$ [11].

4.1 Partial-Coverage Arrays

In most cases, except perhaps for 2D arrays, it is hard to deploy truly regular speaker arrays. Physical constraints limit the placement of speakers, 3D arrays need speakers above and below the listener, and smaller arrays usually have less density of speakers in the vertical direction. Most practical 3D arrays are domes or stacked rings with no speakers below the listener area. The ADT implements several design techniques that produce decoders that perform well for these partial-coverage loudspeaker arrays:

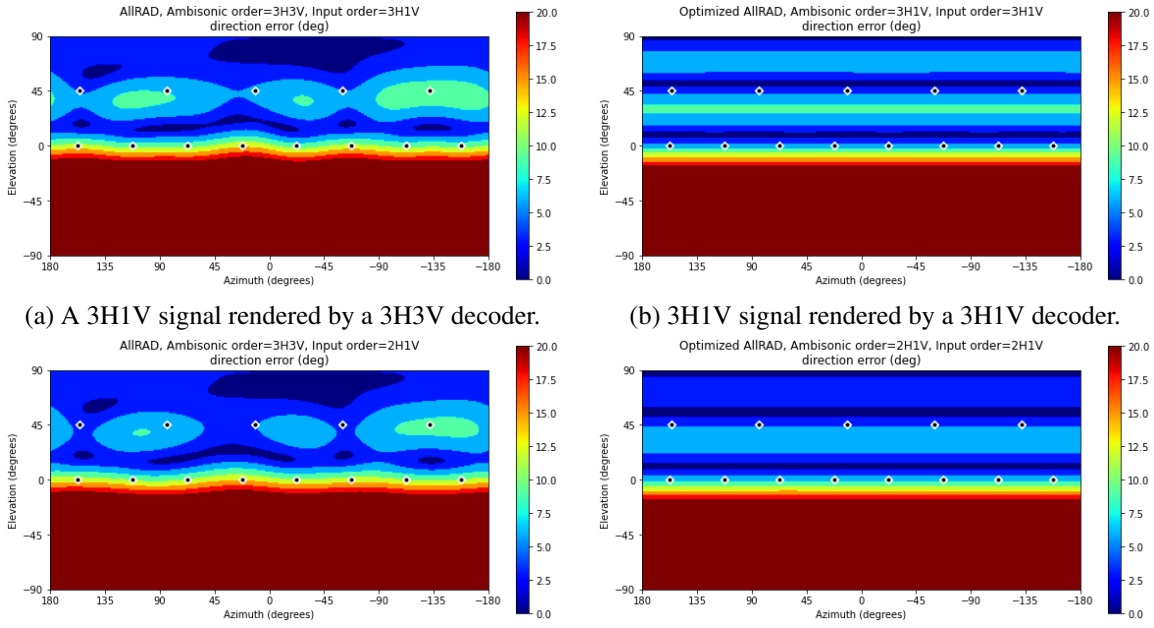
- Use an inversion technique suited to ill-conditioned matrices.
- Derive a new set of basis functions for which inversion is well behaved, EPAD [15].
- Invert a well-behaved full-sphere virtual speaker array, map to a real array, AllRAD [16].

Currently, the AllRAD method is able to cope well with partial-coverage arrays and is our "go-to" technique to design decoders for them.

In general, these techniques trade off localization accuracy for uniform loudness. Typically, $\mathbf{r_E}$ and $\mathbf{r_V}$ will not point in the same direction and localization quality degrades in areas of low density of speakers. These tradeoffs are determined by the particular technique in use and are not directly under a user's control. Many, starting with Gerzon [13], have turned to numerical optimization techniques to enable more direct control over these tradeoffs. The literature is full of descriptions of implementations, some taking days to find a solution, but, as far as we know, the implementation described in the next section is the first one released publicly that is capable of producing two-band decoders.

5 Optimizing Decoders

To address these shortcomings, we implemented a decoder-matrix optimizer that directly implements the above Ambisonic decoder evaluation criteria in its objective function. Each of the criteria is evaluated at 5200 points of a spherical design [17], spatially weighted according to the loudspeaker coverage area, then summed to produce the value of the objective function. Because some of the criteria are non-linear, we employ a constrained, quasi-Newton method, L-BFGS [18], which is available in SciPy's optimization module. Additionally, we employ the JAX library for automatic differentiation of Python/NumPy code to perform the gradient calculation needed by L-BFGS[19]. This provides a large speed-up over finite-difference techniques and will use a GPU if available. As is standard practice in non-linear optimization problems, we add a Tikhonov regularization term to prevent the optimizer from getting stuck in otherwise flat parts of the objective function.



(a) A 3H1V signal rendered by a 3H3V decoder.

(b) 3H1V signal rendered by a 3H1V decoder.

(c) A 2H1V signal rendered by a 3H3V decoder.

(d) 2H1V signal rendered by a 2H1V decoder.

Figure 1: The effect on directional accuracy using a generic 3H3V decoder vs. decoders designed for the specific signal set in use. The speaker array has 8 horizontal and 5 height speakers. The dots show the locations of the loudspeakers.

With large arrays, we find that using an existing matrix for the initial guess ($\mathbf{x_0}$), for example an AllRAD design, ensures quick convergence. For smaller arrays, the initial guess can be a random matrix. Running time for small arrays is a few seconds, larger arrays can take a couple of minutes.

5.1 Optimizing $\mathbf{r_E}$ and $\mathbf{r_V}$ for Mixed-Order Signal Sets

For full-order systems, there are closed-form expressions for the maximum achievable magnitude of $\mathbf{r_E}$ for a particular Ambisonic order such as Table 3.5 in [20], but none (that we know of) exist for mixed-order signal sets. Using too large a value in the objective function makes the convergence behavior less robust. Our solution is to design a mixed-order decoder for a spherical-design 240-loudspeaker array with the desired mixed-order signal set. Due to the integration properties of a spherical design, this is a well-behaved optimization problem and yields an optimal decoder matrix. We then compute $\mathbf{r_E}$ for this matrix at each point in a 5200-point spherical design and use those values as the goal for each corresponding direction in the optimization process for the actual loudspeaker array.

In a second step, the low-frequency matrix is optimized with the goal that $\mathbf{r_V}$ points in the same direction as $\mathbf{r_E}$, $\hat{\mathbf{r_E}} = \hat{\mathbf{r_V}}$ and has a magnitude of 1 over the area covered by the speaker array, $|\mathbf{r_V}| = 1$, thereby satisfying criteria (5) and (2), respectively.

5.2 Tikhonov Regularization Problems

Initial tests showed that the while the Tikhonov regularization term sped up convergence, it also tended to shut off loudspeakers, using the minimum needed for the signal set. While this may be desirable for "spectral impairment" considerations [21], we have found that keeping those speakers active increases the size of the sweet spot [22]. Another consideration is that some arrays, such as the Stage array at CCRMA, use a mixture of speaker types, where some speakers are full-range, while others have limited bass response and power handling capability. We added an optional, per-speaker "spareness penalty" to the objective function to allow a user to specify that some

Figure 2: The Stage at CCRMA. This is a permanent installation of 56 full-range loudspeakers.

speakers should not be turned off by the optimizer.

The left pane of Figure 3 shows how speakers 1, 2, 7, and 8 in the Stage speaker array (four of the 8 big speakers in the ear-level layer) are being turned off by the optimizer when the sparseness penalty is 0. When setting sparse penalty to 1.0 (right pane) those speakers are again active and contributing to the decoded sound field.

6 Results and Discussion

We studied several loudspeaker arrays and found that in each case, the optimizer produced equal to or better-performing decoders than standard techniques such as AllRAD and EPAD. In the following sections, we show examples based on a large and small array.

6.1 The Stage at CCRMA

The Stage, shown in Figure 2, is a small concert space at CCRMA, Stanford University. It has a permanently-mounted array of 56 full-range speakers and eight subwoofers. Of the twenty speakers comprising the main "ear-level" ring, eight are larger and located in movable tower stands. The ear-level ring and the upper speakers form a fairly uniform dome of 48 speakers. There are eight additional speakers at floor level, on the bottom of the eight main tower stands, that were added to anchor the decoded sound image so that sounds coming from the horizontal plane are not elevated. The eight towers also house the subwoofers.

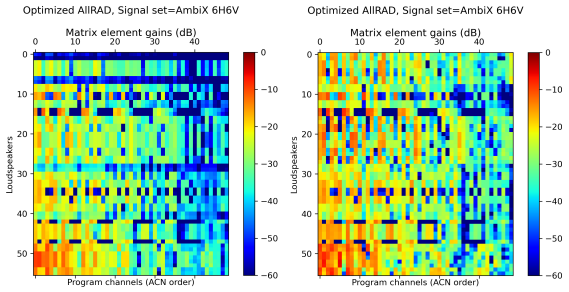


Figure 3: Stage 6th order matrix, sparseness_penalty=0 in left plot, 1.0 in right plot

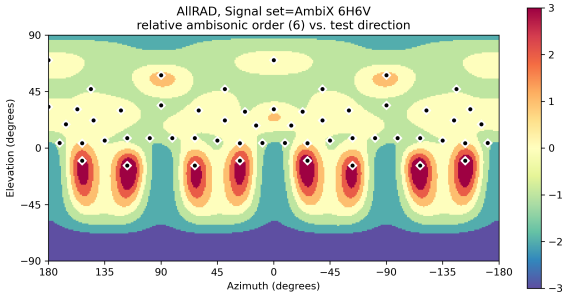
As noted, the 48 upper speakers are distributed uniformly, but the eight lower ones form a significantly sparser ring when compared to the rest of the array, and are not evenly spaced in the horizontal plane. This is a challenge to existing decoder design methods. Figure 4(a) shows the relative Ambisonic order of a 6H6V AllRAD decoder.[1]

The performance in the region just below the horizon which is rendered mainly by these eight speakers is very uneven. The areas of high $|\mathbf{r_E}|$ indicate that sound will tend to jump from one speaker to the next as it is panned around the array and hence, they do not assist in smoothing the performance in the vicinity of the horizontal plane.

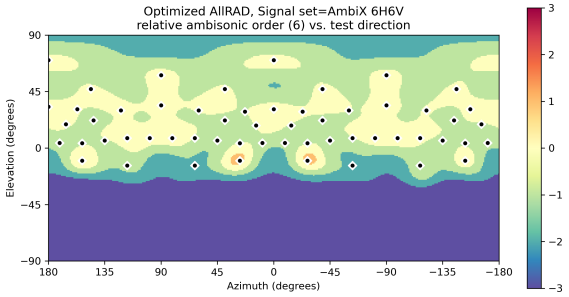
An optimized decoder (Figure 4(b), sparseness_penalty=0.5) trades off performance at low elevations for a smooth integration of the lowest eight speakers into the full array. Note how there is a wider vertical band around the main ring in which the desired order of decoding happens correctly.

The sparseness_penalty parameter can be used to exert some control over how the speakers are used by the

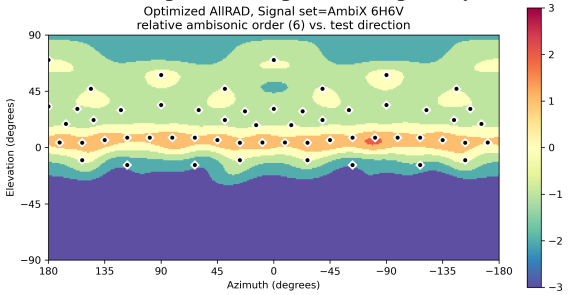
[1] In these plots, the value of rE is calculated, then mapped to the nearest corresponding Ambisonic order and then displayed relative to the design order of the decoder. Hence "0" indicates the decoder is operating at its designed order, "+1" is one up, "-1" one less and so forth, this shows how good the rendering quality is in different directions



(a) 6H6V AllRad



(b) 6H6V optimized, sparseness_penalty=0.5

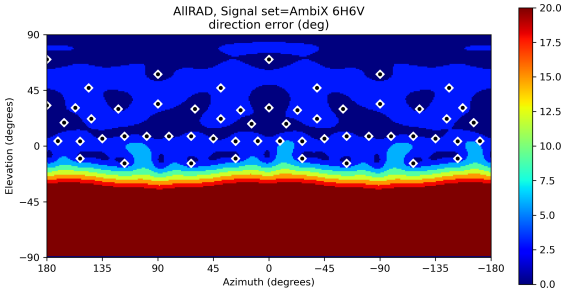


(c) 6H6V optimized, sparseness_penalty=1.0

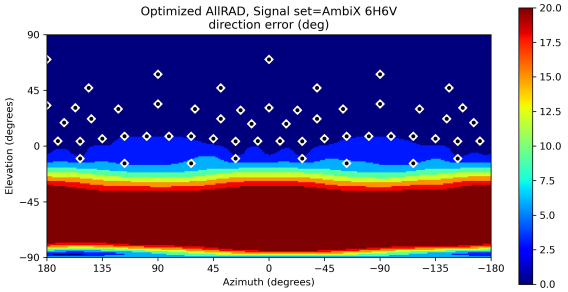
Figure 4: Stage array, reproduction quality relative to sixth order.[1]

optimization process. For this array, setting it to "1" enhances the horizontal performance of the array at the cost of reduced performance at high elevations (Figure 4(c), sparseness_penalty=1).

Figure 5 shows the direction error for both the (a) AllRAD and (b) optimized decoder. In this case the improvement is marginal as the original decoder is already performing very well.



(a) 6H6V AllRad



(b) 6H6V optimized

Figure 5: Stage array, direction error relative to intended direction (clipped at 20 degrees).

6.2 Home Dome 8+5 speaker array

This is a small, two-ring speaker array with an ear-level ring of eight speakers and an upper ring of five smaller speakers at 45 degree elevation. It is a good example of an array that needs an optimized mixed-order decoder.

The lower-eight speaker ring can render third-order horizontal, but the combined array can only do mixed-order rendering.

Figure 6: Home Dome, eight horizontal and five height speakers. Speaker positions highlighted.

Figure 7(a) shows the relative order performance of an AllRAD 3H2V mixed-order decoder for this array. An optimized mixed-order decoder, Figure 7(b), creates a much more even rendering of the sound field, with higher performance in the horizontal plane and even performance in the dome above the listener. Performance suffers at the top of the dome, as is to be expected because of the lower speaker density there.

Direction error is also minimized by the optimized decoder (Figure 8). The improvement is significant in this small array.

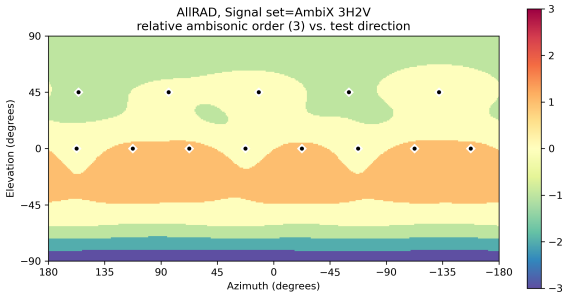
Another way of looking at the directional performance is to plot the actual directions from which sources would originate if moving along lines of constant azimuth or elevation. We can see these plots in Figure 9 which show that the optimized decoder has much less directional error than the plain AllRAD.

Figure 10 shows how adding the second optimization stage for $\mathbf{r_V}$ and using a two-matrix "Vienna" style decoder minimizes the directional mismatch between low- and high-frequency performance. In addition to pointing in the correct direction, the magnitude of $\mathbf{r_V}$ has made uniformly 0.95-1.0 (the ideal value), from a starting point that varied between 0.5 and 1.5 depending on direction.

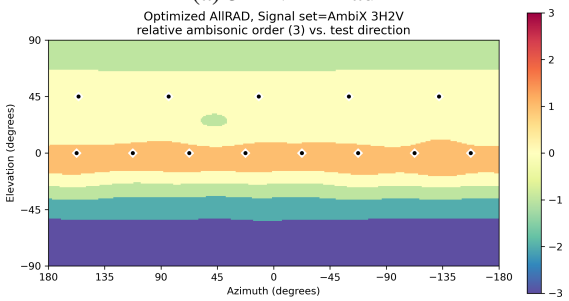
7 Summary

We describe a new implementation of the ADT which has tools that more perfectly and quickly optimize the array performance for the Ambisonic criteria. These tools are applied to the design of two loudspeaker arrays, a 56-loudspeaker professional installation, and a 13-loudspeaker array in a domestic installation. The analysis tools were applied to the question of whether it is acceptable to use a full-order decoder with a mixed-order signal set. Analysis shows that in every case it is better to derive a separate mixed-order decoder.

Early work on Ambisonics described decoders for either 2D or 3D regular arrays of loudspeakers. Most applications in the real world involve arrays that are either irregular or incomplete. One example is approximate hemispherical arrays. Such arrays are inherently irregular and the missing

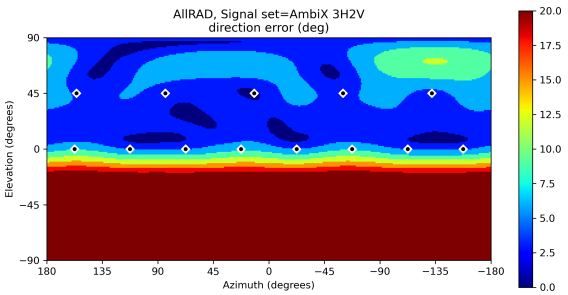


(a) 3H2V AllRad

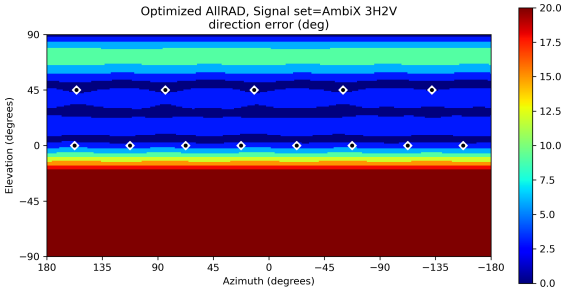


(b) 3H2V optimized

Figure 7: Home Dome decoders, reproduction quality relative to third order.

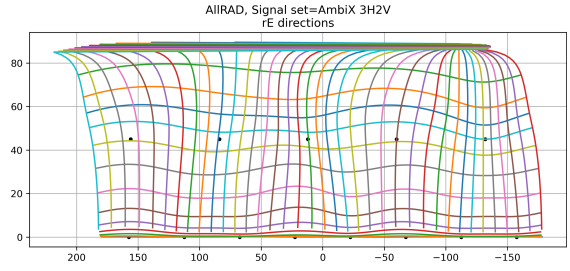


(a) 3H2V AllRad

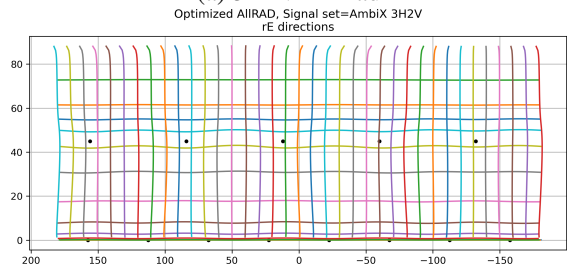


(b) 3H2V optimized

Figure 8: Home Dome decoders, direction error relative to intended direction (clipped at 20 degrees).

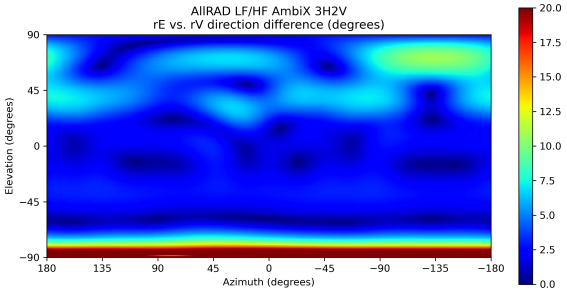


(a) 3H2V AllRad

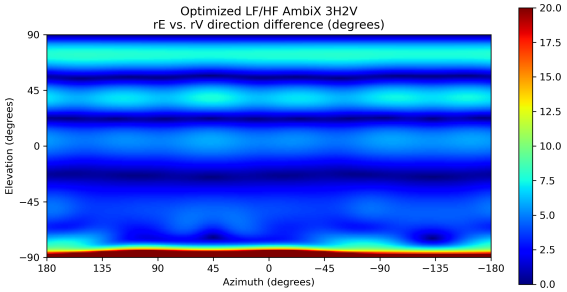


(b) 3H2V optimized

Figure 9: Home Dome decoders, rendered position of source moving along lines of constant azimuth and elevation.



(a) 3H2V AllRad



(b) 3H2V optimized

Figure 10: Home Dome decoder, $\mathbf{r_E}$ vs. $\mathbf{r_V}$ direction error.

bottom half makes the resulting decoder have increasing error at and below the horizon. There is a second problem having to do with mixed-order decoders. It is frequently the case that the density of loudspeakers is less for directions above the horizontal. This happens either because of the expense of the additional loudspeakers or because of difficulties mounting the speakers. In this case, the array has a different capability in different directions, almost always with greater performance for the horizontal direction than for height. Also, some microphone arrays have different Ambisonic order performance in horizontal and vertical directions.

Subsequent work described methods such as AllRAD for deriving decoders for these arrays. These methods result in decoders that only approximately meet the Ambisonic criteria. Numerical optimization methods can be used to enable the decoder to have nearly perfect behavior through sparse regions and at the edge of array coverage. By "nearly perfect" we mean as good as possible for the number of loudspeakers available. The visualization tools provided with the ADT enable the designer to make choices as to which ambisonic parameters are to be optimized. These can be used, for instance, to determine how many of a given total number of loudspeakers are to be used for horizontal and how many for height.

Informal listening tests were performed for the two loudspeaker arrays described above but were limited to only a single listener due to the COVID-19 restrictions at Stanford and SRI. Future work will include more formal tests with a larger set of listeners.

The code described in this paper is free and open source, and can be accessed via the ADT repository[1]. The implementation is a work in progress, but is fully capable of producing working (and very good) decoders. The code can be run via Jupyter notebooks using Google Research's Colaboratory facility [23]. We provide notebooks that reproduce the results in this paper. The FAUST code produced can be compiled online as well, producing plugins for most types of audio processing programs [24].

References

[1] Heller, A. J., "The Ambisonic Decoder Toolbox (ADT)," `https:`

//bitbucket.org/ambidecodertoolbox/adt.git, 2011-2021, [Online; accessed 31-March-2021].

[2] Travis, C., "A New Mixed-Order Scheme for Ambisonic Signals," in *Proc. 1st Ambisonics Symposium*, pp. 1–6, 2009.

[3] Heller, A. and Benjamin, E. M., "Design and implementation of filters for Ambisonic decoders," in *1st International Faust Conference (IFC-18)*, pp. 1–6, Mainz, 2018.

[4] Benjamin, E. M., "A Second-Order Soundfield Microphone with Improved Polar Pattern Shape," in *AES 133rd Convention*, San Francisco, 2012.

[5] "Core Sound OctoMic™ 2nd-order Ambisonic Microphone," https://www.core-sound.com/products/octomic, 2018, [Online; accessed 23-Apr-2021].

[6] "Voyage Audio Spatial Mic," https://voyage.audio/spatialmic/, 2019, [Online; accessed 23-Apr-2021].

[7] "Brahma Studio 8," https://brahmamic.com/products/brahma-studio-8/, 2020, [Online; accessed 23-Apr-2021].

[8] "Reynolds Microphones, Eight-Capsule A-Type," https://www.facebook.com/reynoldsmicrophones/photos/2102762213120167, 2018, [Online; accessed 23-Apr-2021].

[9] Lopez-Lezcano, F., "The SpHEAR project update: Refining the OctaSpHEAR, a 2nd order ambisonics microphone," in *EAA Spatial Audio Signal Processing Syposium*, pp. 103–108, Paris, 2019.

[10] Frank, M., *Phantom Sources using Multiple Loudspeakers in the Horizontal Plane* , Ph.D. thesis, Institute of Electronic Music and Acoustics, University of Music and Performing Arts, Graz, 2013.

[11] Gerzon, M. A., "General Metatheory of Auditory Localisation," in *92nd Audio Engineering Society Convention Preprints*, 3306, Vienna, 1992.

[12] Scharine, A. and Letowski, T., *Auditory Conflicts and Illusions*, pp. 579–598, U.S. Army Aeromedical Research Laboratory, 2009, ISBN 978-0-615-28375-3, doi:10.13140/2.1.3684.4804.

[13] Gerzon, M. A. and Barton, G. J., "Ambisonic Decoders for HDTV," in *92nd Audio Engineering Society Convention Preprints*, 3345, Vienna, 1992.

[14] Smith, J. O., "Audio Signal Processing in FAUST," 2013, online, accessed 1-Feb-2014.

[15] Zotter, F., Pomberger, H., and Noisternig, M., "Energy-Preserving Ambisonic Decoding," *Acta Acustica united with Acustica*, 98(1), pp. 37–47, 2012.

[16] Zotter, F. and Frank, M., "All-Round Ambisonic Panning and Decoding," *Journal Of The Audio Engineering Society*, 60(10), pp. 807–820, 2012.

[17] Wikipedia contributors, "Spherical design — Wikipedia, The Free Encyclopedia," https://en.wikipedia.org/w/index.php?title=Spherical_design&oldid=1004415845, 2021, [Online; accessed 23-May-2021].

[18] Wikipedia contributors, "Limited-memory BFGS — Wikipedia, The Free Encyclopedia," https://en.wikipedia.org/w/index.php?title=Limited-memory_BFGS&oldid=996198021, 2020, [Online; accessed 23-May-2021].

[19] "JAX: Composable transformations of Python+NumPy programs: differentiate, vectorize, JIT to GPU/TPU, and more," https://opensource.google/projects/jax, 2020, [Online; accessed 23-Apr-2021].

[20] Moreau, S., "Étude et réalisation d'outils avancés d'encodage spatial pour la technique de spatialisation sonore Higher Order Ambisonics : microphone 3D et contrôle de distance," *PhD Thesis*, 2006.

[21] Solvang, A., "Spectral impairment for two-dimensional higher order ambisonics," *Journal of the Audio Engineering Society*, 56(4), pp. 267–279, 2008.

[22] Benjamin, E. M. and Heller, A., "Assessment of Ambisonic System Performance Using Binaural Measurements," in *Audio Engineering Society Convention 137*, 2014.

[23] "Welcome To Colaboratory," https://colab.research.google.com/notebooks/intro.ipynb, 2020, [Online; accessed 23-Apr-2021].

[24] "Faust Editor," https://fausteditor.grame.fr, 2015, [Online; accessed 23-Apr-2021].