Ambisonics Navigation (AmbiNav) Toolkit

Joseph G. Tylka
josephgt@princeton.edu

v0 – October 20th, 2018

Summary

The ambisonics navigation (AmbiNav) toolkit is an open-source collection of MATLAB functions for performing virtual navigation of higher-order ambisonics recordings. This document describes the methods implemented in the toolkit and provides instructions for its use.

1 Introduction

This document is structured as follows. In Section 2, we describe the mathematical conventions followed in this toolkit. Finally, in Section 3, we describe the various functions included in the toolkit and how to use them.

1.1 Contents

The toolkit consists of the following items:

1. the core set of MATLAB functions for the toolkit (`/AmbiNav_*.m`);
2. the source LaTeX files for this manual (`/doc/`);
3. a MATLAB file containing examples of how to use the toolkit (`/examples.m`); and
4. MATLAB implementations of some published virtual navigation methods (`/methods/`).

2 Conventions

Here, we use real-valued spherical harmonics as given by Zotter [1, section 2.2]

$$Y_l^m(\theta,\phi) = N_l^{|m|} P_l^{|m|}(\sin\theta) \times \begin{cases} \cos m\phi & \text{for } m \geq 0, \\ \sin |m|\phi & \text{for } m < 0, \end{cases}$$

where P_l^m is the associated Legendre polynomial of degree l and order m, as defined in the MATLAB `legendre` function by[1]

$$P_l^m(x) = (-1)^m (1-x^2)^{m/2} \frac{d^m}{dx^m} P_l(x), \qquad \text{with} \qquad P_l(x) = \frac{1}{2^l l!}\left[\frac{d^l}{dx^l}(x^2-1)^l\right],$$

[1]See: `https://www.mathworks.com/help/matlab/ref/legendre.html`

and N_l^m is a normalization term which, for the orthonormal (N3D) spherical harmonics with Condon-Shortley phase,[2] is given by [2]

$$N_l^m = (-1)^m \sqrt{\frac{(2l+1)(2-\delta_m)}{4\pi}\frac{(l-m)!}{(l+m)!}},$$

where δ_m is the Kronecker delta. With an inner product defined by integrating over all directions, the squared-norm of these spherical harmonics is thus given by

$$\|Y_l^m\|^2 = 1.$$

We also adopt the ambisonics channel numbering (ACN) convention [2] such that for a spherical harmonic function of degree $l \in [0, \infty)$ and order $m \in [-l, l]$, the ACN index n is given by $n = l(l+1) + m$ and we denote the spherical harmonic function by $Y_n \equiv Y_l^m$.

3 MATLAB Functions

The core MATLAB functions included in the toolkit are listed and described in Table 1.

Acknowledgements

This toolkit requires the 3D3A Lab's MATLAB Toolbox[3] by R. Sridhar and J. G. Tylka.

References

[1] Franz Zotter. *Analysis and Synthesis of Sound-Radiation with Spherical Arrays.* PhD thesis, University of Music and Performing Arts Graz, 2009.

[2] Christian Nachbar, Franz Zotter, Etienne Deleflie, and Alois Sontacchi. ambiX - A Suggested Ambisonics Format. In *Proceedings of the 3rd Ambisonics Symposium*, June 2011. URL `http://ambisonics.iem.at/proceedings-of-the-ambisonics-symposium-2011/ambix-a-suggested-ambisonics-format`.

[3] Joseph G. Tylka and Edgar Y. Choueiri. Soundfield Navigation using an Array of Higher-Order Ambisonics Microphones. In *Audio Engineering Society Conference: 2016 AES International Conference on Audio for Virtual and Augmented Reality*, September 2016. URL `http://www.aes.org/e-lib/browse.cfm?elib=18502`.

[4] Nail A. Gumerov and Ramani Duraiswami. *Fast Multipole Methods for the Helmholtz Equation in Three Dimensions.* Elsevier Science, 2005.

[2] Note that including Condon-Shortley phase in the normalization term cancels it in the associated Legendre term.
[3] Available online: `https://github.com/PrincetonUniversity/3D3A-MATLAB-Toolbox`

Function	**Description**
`AmbiNav_ArraySpacing`	Computes the (approximate) spacing between HOA microphones.
`AmbiNav_CoefficientA`	Returns the recurrence coefficient a for spherical harmonic translation [1, Eq. (145)].
`AmbiNav_CoefficientB`	Returns the recurrence coefficient b for spherical harmonic translation [1, Eq. (146)].
`AmbiNav_InterpolationFilters`	Computes regularized least-squares interpolation filters as defined by Tylka and Choueiri [3].
`AmbiNav_InterpolationWeights`	Computes linear interpolation weights for specified grid points and query points.
`AmbiNav_KDThreshold`	Returns the pre-defined minimum translation distance, approximately 1 mm at 10 Hz.
`AmbiNav_Pitch90`	Returns the ambisonics rotation matrix for 90° pitch.
`AmbiNav_PlaneWaveTranslation`	Computes plane-wave translation coefficients for a specified grid of directions and a specified translation vector.
`AmbiNav_Roll90`	Returns the ambisonics rotation matrix for 90° roll.
`AmbiNav_SphericalHarmonicY`	Computes ACN/N3D real-valued spherical harmonics; returns a matrix of spherical harmonics up to a specified ambisonics order and for a specified grid of positions.
`AmbiNav_Start`	Starts the AmbiNav Toolkit and the 3D3A MATLAB Toolbox.
`AmbiNav_Translation`	Computes ambisonics translation coefficients for a specified translation vector.
`AmbiNav_TriangulateSource`	Estimates the source position given microphone positions and corresponding directions-of-arrival.
`AmbiNav_Yaw90`	Returns the ambisonics rotation matrix for 90° yaw.
`AmbiNav_YawRotation`	Returns the ambisonics rotation matrix for a specified yaw.
`AmbiNav_ZRotation`	Returns the ambisonics rotation matrix to align the z-axis to a given direction.
`AmbiNav_ZTranslation`	Returns the ambisonics translation coefficients for translation along the z-axis.

Table 1: Core MATLAB functions in the toolkit.

Function	**Description**
`gumerov2005`	Ambisonics navigation using translation coefficients [1, 4].
`schultz2013`	Ambisonics navigation using plane-wave translation [5].
`southern2009`	Ambisonics navigation using linear interpolation [6].
`thiergart2013`	Ambisonics navigation via sound field analysis and modeling [7].
`thiergart2013_analysis`	Time-frequency domain analysis of a sound field [7].
`thiergart2013_synthesis`	Ambisonics rendering of a modeled sound field [7].
`tylka2016`	Ambisonics navigation using least-squares interpolation filters [3].

Table 2: Virtual navigation methods implemented in the toolkit.

[5] Frank Schultz and Sascha Spors. Data-Based Binaural Synthesis Including Rotational and Translatory Head-Movements. In *Audio Engineering Society Conference: 52nd International Conference: Sound Field Control - Engineering and Perception*, September 2013. URL `http://www.aes.org/e-lib/browse.cfm?elib=16894`.

[6] Alex Southern, Jeremy Wells, and Damian Murphy. Rendering walk-through auralisations using wave-based acoustical models. In *Signal Processing Conference, 2009 17th European*, pages 715–719, August 2009.

[7] Oliver Thiergart, Giovanni Del Galdo, Maja Taseska, and Emanuël A. P. Habets. Geometry-Based Spatial Sound Acquisition Using Distributed Microphone Arrays. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(12):2583–2594, December 2013. ISSN 1558-7916. doi: 10.1109/TASL.2013.2280210.