www.ak.tu-berlin.de/AKtools

Fabian Brinkmann, Stefan Weinzierl

Audio Communication Group
Technical University Berlin
Einsteinufer 17c
10587 Berlin - Germany

ak-tools@ak.tu-berlin.de

March 8, 2018

Getting Started

To start using AKtools open Matlab, navigate to the AKtools folder and run

`AKtoolsStart.m`

This will add all needed folders to the end of your Matlab search path. Follow the output in the command window in case you want to save AKtools to your Matlab search path.

AKtools contains scripts that illustrate the usage of all main functions, and can be found in the folder

`1_DemoScripts`

The scripts are named according to their purpose and include a description of their content at the beginning. They are subdivided into blocks that can be evaluated using `ctrl+enter`. The processing steps in each block are commented for documentation.

The demo scripts call one or more functions inside

`2_Tools`

If you want more information on a certain function either type `help functionName` in the Matlab Command Window or open it. For your convenience, each function header contains

- The function definition, e.g.
  `sweep = AKsweepFD(varargin)`
- Exemplary function calls with short explanations, e.g.
  To obtain a perfect sweep with 4096 samples call
  `sweep = AKsweepFD('NFFT', 4096, 'preset', 'perfect')`, and/or
- a link to a demo script inside `1_DemoScripts` (if there is one)
- A list of input and output arguments
- Author, and contact information
- License information
- References for further reading (if available)

AKtools use the "FABIAN head-related transfer function database" from Brinkmann et al. (2016) that is available from

- https://dx.doi.org/10.14279/depositonce-5718.2

Demo contents

`AKplotDemo:` Visualization of audio signals. This can be used to display single impulse responses, magnitude/phase/group-delay spectra, but also is able to visualize data on spherical sampling grids, and show interaural-time and level-differences in head-related impulse responses.

`AKioDemo:` Is short for AK in/out and introduces multi-channel audio playback and recording with sample accuracy to AKtools. For this purpose *pa_wavplay* and *playrec* are used which are Matlab externals that directly use audio libraries of the operating system for best performance.

`AKmeasureDemo:` Shows how to measure impulse responses in Matlab, including sine-sweep generation, deconvolution, and basic post-processing and inspection of the results. It can be used for single or multi channel systems, and tackles averaging multiple measurements for an improved signal-to-noise ratio, and automatically detects and corrects clipping in the recorded signals.
`Further reading:` Müller and Massarani (2001)

`AKtestSignalsDemo:` Generation of some common audio signals that can be used for testing audio equipment, and acoustic measurements. Among others, this includes pink and white noise, as well as a large variate of sine sweeps designed in the time or frequency domain (linear, logarithmic, perfect, arbitrary spectral envelope).
`Further reading:` Müller and Massarani (2001); Antweiler et al. (2011); Novák et al. (2010)

`AKdeconvolutionDemo:` Often, acoustic measurements use deconvolution to obtain impulse responses. The demo shows the use of `AKdeconv` which is able to address some common issues in acoustic measurements such as noise signals, transfer function inversion or compensation of electro-acoustic transducers (e.g. microphones).
`Further reading:` Müller and Massarani (2001)

`AKfilterDemo:` Shows the use of `AKfilter` for filtering audio signals with high/low/band-passes, band-stops, parametric equalizers (PEQs), shelve filters, cross-over networks, and fractional octave filters.
`Further reading:` Zölzer (2002); Bristow-Johnson (1994); Bohn (2005)

`AKroomSimulationDemo:` A binaural simulation of a shoebox shaped room including source and receiver directivity. Uses image sources for calculating early reflections and decaying noise for the late reverberation.
`Further reading:` Brinkmann et al. (2018)

`AKhrirInterpolationDemo:` Shows how to use spherical harmonics for head-related impulse response interpolation, and includes the possibility to interpolate between different head-above-torso orientations.
`Further reading:` Brinkmann et al. (2015)

`AKregulatedInversionDemo:` Inverting transfer functions of electro-acoustic systems is a common, but not necessarily trivial task. This demo shows how to use regulated inversion to invert headphone transfer functions for use in binaural technology. It includes different approaches of the regularization differing in their complexity and quality and handles everything from averaging the transfer functions to saving the generated filters.
`Further reading:` Kirkeby and Nelson (1999); Norcross et al. (2006); Schärer and Lindau (2009); Lindau and Brinkmann (2012)

`AKperceptualMixingTimeDemo:` Estimates the perceptual mixing time, i.e. the instant a room's sound field is percieved as being diffuse. The perceptual mixing time is estimated using model- and/or signal-based predictors. It can be used to split (binaural) room impulse responses into an early part containing directional information, and a late part without directional information.
`Further reading:` Lindau et al. (2012); Abel and Huang (2006)

`AKsphericalHarmonicsDemo:` Gives some examples for generating and manipulating spherical harmonics basis functions.
`Further reading:` Williams (1999); Rafaely (2015)

`AKsphericalHarmonicsTransformDemo:` Shows how to apply the spherical harmonic transform (also termed spherical Fourier transform) to a dataset of head-related impulse responses. It can be used to show the effect of truncation order, and the influence of using complex spectra or magnitude/phase spectra as input for the transform.
`Further reading:` Williams (1999); Rafaely (2015)

`AKsphericalHeadDemo:` Spherical harmonics based model of a spherical head. This is a range depended simulation of the sound field scattered by a rigid sphere.
`Further reading:` Duda and Martens (1998)

`AKsphericalCapExtrapolationDemo:` Gives an example of how to extrapolate missing data on spherical sampling grids based on low order spherical harmonics representation of the data.
`Further reading:` Ahrens et al. (2012)

`AKtoaDemo:` Shows how to detect and remove the time of arrival (TOA) from impulse responses with sub-sample accuracy. This his for example helpful for interpolating impulse responses, for estimating the interaural time difference (ITD), or for ITD estimation.
`Further reading:` Brinkmann et al. (2015); Katz and Noisternig (2014); Lindau et al. (2010)

`AKsubGridDemo:` Makes it possible to quickly select points from spatial sampling grids including horizontal, median, and frontal planes.

`AKphaseManipulationDemo:` Gives examples of how to manipulate the phase of a transfer function and can be used to transform it to minimum, linear, and zero phase.
`Further reading:` Oppenheim et al. (1999)

`AKnormalizationDemo:` Makes it possible to normalize multi channel data in the time and frequency domain.

`AKsphSplineInterpDemo:` Demonstrated the use of spherical spline interpolation which might be useful for instance if a dataset does not meet the requirements for spherical harmonics based transformations.
`Further reading:` Wahba (1981, 1982)

`AKboxplotDemo:` Visualization of distributed data such as ratings from listening tests or results from repeated measurements. Allows grouping of conditions and selection of different box layouts including confidence intervals, percentile ranges, mean and median, and support of subplots.

Acknowledgement

If you use AKtools please cite

> Fabian Brinkmann, Stefan Weinzierl: “AKtools—An Open Software Toolbox for Signal Acquisition, Processing, and Inspection in Acoustics,” *142nd AES Convention*, e-Brief 309, Berlin, Germany, 2017.

Availablility

The latest release of AKtools is available from `www.ak.tu-berlin.de/aktools`, the current state of work can be accessed via subversion

> `https://svn.ak.tu-berlin.de/svn/AKtools`
> *username: aktools, password: ak*

Support

We welcome any feedback on bugs inside AKtools as well as features that could be added. You can contact us via ak-tools@ak.tu-berlin.de or use our bug tracker at

> `https://www2.ak.tu-berlin.de/bug-tracker`

License

AKtools are licensed under the European Union Public License (EUPL) v. 1.1 of which a copy can be found inside AKtools. The license gives you the right to

- use AKtools in any circumstance and for all usage
- reproduce AKtools
- modify the original AKtools, and make derivative works based upon AKtools
- communicate to the public, including the right to make available or display AKtools or copies thereof to the public
- distribute AKtools or copies thereof
- lend and rent AKtools or copies thereof
- sub-license rights in AKtools or copies thereof

Contributions

Alexander Lindau, Andreas Rotter, André Giese, David Ackermann, Daniel Kahlhöfer, Dmitry Grigoriev, Fabian Brinkmann, Frank Schultz, Hannes Helmholz, Jan-Henrik Hanschke, Marc Voigt, Omid Kokabi, Silke Bögelein, Stefan Weinzierl, Zora Schärer Kalkandjiev

Third party code (included)

`pa_wavplay` by Matt Frear & Joseph Desloge; `playrec` by Robert Humphrey, `SOFA Matlab API` by Piotr Majdak *et al.*, parts of the `auditory toolbox` by Malcolm Slaney, `cbfreeze`, `cbhandle` by Carlos Adrian Vargas Aguilera; `makeColormap` by Doug Hall; `mmpolar` by Duane Hanselman; `subtightplot` by Felipe G. Nievinski & Pekka Kumpulainen & Nikolay S.; `j_polynomial` by John Burkardt; This product includes color specifications and designs developed by Cynthia Brewer (`http://colorbrewer.org/`).

Third party code (needed)

`ASIO4all` by Michael Tippach (only needed for Windows computers)
http://www.asio4all.com

References

Abel, Jonathan S and Patty Huang (2006): “A simple, robust measure of reverberation echo density.” In: *121th AES Convention, Convention Paper 6985*. San Francisco, USA.

Ahrens, Jens; Mark R P Thomas; and Ivan J Tashev (2012): “HRTF magnitude modeling using a non-regularized least-squares fit of spherical harmonics coefficients on incomplete data.” In: *APSIPA Annual Summit and Conference*. Hollywood, USA.

Antweiler, Christiane; Aulis Telle; Peter Vary; and Gerald Enzner (2011): “Perfect-sweep NLMS for time-variant acoustic system identification.” In: *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Prague, Czech Republic.

Bohn, Dennis (2005): *Linkwitz Riley crossovers: A primer*. Tech. rep., Rane.

Brinkmann, Fabian; Vera Erbes; and Stefan Weinzierl (2018): “Extending the closed form image source model for source direct.” In: *Fortschritte der Akustik – DAGA 2018*. Munich, Germany.

Brinkmann, Fabian; Reinhild Roden; Alexander Lindau; and Stefan Weinzierl (2015): “Audibility and interpolation of head-above-torso orientation in binaural technology.” In: *IEEE J. Sel. Topics Signal Process.*, **9**(5), pp. 931–942.

Brinkmann, Fabian; et al. (2016): “The FABIAN head-related transfer function data base.” http://depositonce.tu-berlin.de/handle/11303/6153.

Bristow-Johnson, Robert (1994): “The equivalence of various methods of computing biquad coefficients for audio parametric equalizers.” In: *97th AES Convention, Preprint*. San Francisco, USA.

Duda, Richard O. and William L. Martens (1998): “Range dependence of the response of a spherical head model.” In: *J. Acoust. Soc. Am.*, **104**(5), pp. 3048–3058.

Katz, Brian F. G. and Markus Noisternig (2014): “A comparative study of interaural time delay estimation methods.” In: *J. Acoust. Soc. Am.*, **135**(6), pp. 3530–3540.

Kirkeby, Ole and Philip A. Nelson (1999): “Digital filter design for inversion problems in sound reproduction.” In: *J. Audio Eng. Soc.*, **47**(7/8), pp. 583–595.

Lindau, Alexander and Fabian Brinkmann (2012): “Perceptual evaluation of headphone compensation in binaural synthesis based on non-individual recordings.” In: *J. Audio Eng. Soc.*, **60**(1/2), pp. 54–62.

Lindau, Alexander; Jorgos Estrella; and Stefan Weinzierl (2010): “Individualization of dynamic binaural synthesis by real time manipulation of the ITD.” In: *128th AES Convention, Convention Paper*. London, UK.

Lindau, Alexander; Linda Kosanke; and Stefan Weinzierl (2012): “Perceptual evaluation of model- and signal-based predictors of the mixing time in binaural room impulse responses.” In: *J. Audio Eng. Soc.*, **60**(11), pp. 887–898.

Müller, Swen and Paulo Massarani (2001): “Transfer function measurement with Sweeps. Directors’s cut including previously unreleased material and some corrections.” In: *J. Audio Eng. Soc. (Original release)*, **49**(6), pp. 443–471.

Norcross, Scott G.; Martin Bouchard; and Gilbert A. Soulodre (2006): "Inverse Filtering design using a minimal phase target function from regularization." In: *121th AES Convention, Convention Paper 6929*. San Francisco, USA.

Novák, Antonín; Laurent Simon; František Kadlec; and Pierrick Lotton (2010): "Nonlinear system identification using exponential swept-sine signal." In: *IEEE Trans. on Instrumentation and Measurement*, **59**(8), pp. 2220–2229.

Oppenheim, Alan V; Ronald W Schafer; and John R. Buck (1999): *Discrete-time signal processing*. 2nd edition. Upper Saddle, USA: Prentice Hall.

Rafaely, Boaz (2015): *Fundamentals of spherical array processing*, vol. 8 of *Springer Topics in Signal Processing*. 1. Berlin, Heidelberg, Germany: Springer.

Schärer, Zora and Alexander Lindau (2009): "Evaluation of Equalization Methods for Binaural Signals." In: *126th AES Convention, Convention Paper*. Munich, Germany.

Wahba, Grace (1981): "Spline interpolation and smoothing on the sphere." In: *SIAM J. Sci. Stat. Comput.*, **2**(1), pp. 5–16.

Wahba, Grace (1982): "Erratum: Spline interpolation and smoothing on the sphere." In: *SIAM J. Sci. Stat. Comput.*, **2**(2), pp. 385–386.

Williams, E G (1999): *Fourier Acoustics. Sound radiation and nearfield acoustical holography*. 1. Academic Press.

Zölzer, Udo (2002): *DAFX:Digital Audio Effects*. John Wiley & Sons.