The AMT Team:
Piotr Majdak, Roberto Barumerli, Alejandro Osses, Michael Mihocic, Joonas Guevara, Robert Baumgartner

The 3rd AMT Workshop

Turin, 8.9.2023
The AMT Workshop: Structure and Goals

- What is the AMT and how can it be useful for you?
  - General AMT structure
  - The core functionality
  - Basic: Using AMT models
  - Advanced: Make your own model
  - Hands-on in two groups (Basic, Advanced)

- Goals:
  - Do you know what is the AMT and its components?
  - Do you have a basic idea of what you can do with the AMT?
  - Basic: Are you able to install the AMT and run its models?
  - Advanced: Are you able to implement your own model?
The AMT

- An open-source and open-access toolbox for auditory modeling
  - A tool for reproducing scientific experiments with auditory models
  - A framework for developing new models
- Environment:
  - Core: Matlab/Octave
  - Models: (nearly) any programming language
- Software license:
  - Core: GPL version 3
  - Models: Multi licensing; (nearly) any GPL3-compatible license
- Link of a large amount of data and complex code
- Integration of other toolboxes useful in auditory research
The AMT

- 2009: Development initiated by Peter Søndergaard
- 2012: Development moved to ÖAW
- 2021: AMT 1.0 released
- Now: Over 60 auditory models, 40 GB of auditory data
- Two releases per year, with a focus on sustainability

- Community work: Most of the models “donated” to the AMT
- Increase the authors’ visibility and add citations
Typical stages of auditory models

Left ear

- Outer ear
- Middle ear
- Basilar membrane
- Hair cells
- Auditory nerve
- Brain-stem

Right ear

Binaural

Perception & Behavior
Typical stages of auditory models

- Model ≠ Model implementation
- Models need data
- Model implementations often rely on toolboxes
- Some models use common functionality, some are stand-alone concepts
The Auditory Modeling Toolbox (AMT)

- Model ≠ Model implementation
- Models need data
- Model implementations often rely on toolboxes
- Some models use common functionality, some are stand-alone concepts
The Auditory Modeling Toolbox (AMT)

- Model ≠ Model implementation: Smith et al. (2022) → smith2022
The Auditory Modeling Toolbox (AMT)

- Model ≠ Model implementation: Smith et al. (2022) → smith2022
- Models need data: Auxiliary data
  - Dedicated function: → data_smith2022
  - Direct: → amt_load('smith2022', ..)
Model implementations often rely on toolboxes: Third-party toolboxes

- Large Time Frequency Analysis Toolbox (LTFAT)
- **Optional:** Spatially Oriented Format for Acoustics (SOFA) Toolbox, Sound Field Synthesis (SFS) Toolbox, Circular Statistics (CircStat) Toolbox, Binaural Spherical Harmonics (BinSH) Toolbox
The Auditory Modeling Toolbox (AMT)

- Core functions: the main power of the AMT
  - Storing pre-calculated data: Cache \(\rightarrow\) `amt_cache`
  - Data handling: Auxiliary data \(\rightarrow\) `amt_load`
  - Simulation of experiments: \(\rightarrow\) `amt_emuexp`
The Auditory Modeling Toolbox (AMT)

- **Models**
  - breebaart2011

- **Model stages:**
  - breebaart2011_outmiddlefilter
  - breebaart2011_eicell
  - breebaart2011_centralproc
The Auditory Modeling Toolbox (AMT)

- **Common functions:**
  - `scaletodb SPL`
  - `erb2fc`
  - `middleearfilter`
  - `gammatone`
  - `itdestimator`
  - `adaptloop`
The Auditory Modeling Toolbox (AMT)

- Demonstrations: Simple scripts demonstrating the functionality
  → demo_smith2022
Demonstrations

demo_hohmann2002;
Demonstrations

demo_chen2011;
Demonstrations

demo_reijniers2014;
The Auditory Modeling Toolbox (AMT)

- Demonstrations: Simple scripts demonstrating the functionality
  → demo_smith2022
- Experiments: Functions reproducing published results
  → exp_smith2022('fig3')
Experiments

Screenshot from Figure 9
(Baumgartner et al., 2014):

Experiments

Screenshot from Figure 9 (Baumgartner et al., 2014):

Output of exp_baumgartner2014('fig9');

Experiments

Output of
exp_osses2022('fig4');

- Blue: Dau et al. (1997)
- Orchid: Osses and Kohlrausch (2021)
- Green: Zilany et al. (2014)
- Maroon: Bruce et al. (2018)
- Red: Verhulst et al. (2015)
- Skyblue: Verhulst et al. (2018)
- Magenta: King et al. (2019)
- Black: Relano-Iborra et al. (2019)

The Auditory Modeling Toolbox (AMT)

- Code: Files structured in directories
- Special directories:
  - auxdata
  - cache
  - defaults
  - mat2doc
Documentation

- Website: http://amtoolbox.org

THE AUDITORY MODELING TOOLBOX

Select the AMT version:

- 1.5.0 (July 2023). See changes.
- 1.4.0 (May 2023). See changes.
- 1.3.0 (March 2023). See changes.
- 1.2.0 (May 2022). See changes.
- 1.1.0 (Dec 2021). See changes.
- 1.0.0 (May 2021). See changes.
- 0.10.0 (May 2020). See changes.
- 0.9.9 (September 2017). See changes.
- 0.9.8 (June 2017). See changes.
- 0.9.7 (June 2016)
- 0.9.6 (June 2014)
- 0.9.5 (March 2014)
- 0.9.2 (October 2013)

Bug or typo?

If you think you have found a bug in the code or a typo in the documentation, check out the bug tracker and input the description of the issue you have found. The bug tracker contains a list of all current issues within the AMT and all feature requests for the future.
AMT - Online documentation

General

This is the most complete, and up-to-date description of the AMT. This documentation is directly included in the M-files and it is auto-generated for this website. Because of the automatic generation, the appearance on the website may suffer some details. Note that the current status of the models can be found in the section Models at this website.

New to the AMT?

Download the AMT full package, which provides all third-party toolboxes, start the AMT with `amt_start` and compile the binaries with `amt_mex`. To stop the session, use `amt_stop` which removes the added paths and resets the configuration.

Note that the installation can be also done by calling `amt_start('install')`, which downloads the third-party toolboxes (if missing) and triggers compilation of the binaries.

Important files in the AMT base path:
AMT - Core functions

Installation and session management

- `amt_mex` - Compile binaries (Mex/Oct interfaces)
- `amt_stop` - Stop the AMT session and remove all AMT paths

General functions

- `amt_cache` - Cache variables for later or retrieves variables from cache
- `amt_disp` - AMT-specific overload of the function 'disp'
- `amt_emuexp` - Emulate psychoacoustic experiments
- `amt_flags` - Return the start-up flags of the AMT
- `amt_load` - Load auxiliary data of a model
- `amt_extern` - Execute functions in an external environment (e.g., Python)

Paths and configuration

- `amt_configuration` - Return the configuration of the AMT
AMT_CONFIGURATION - Get and set the configuration of the current AMT session

Usage:

```python
[flags, kv] = amt_configuration;
[flags, kv] = amt_configuration('cacheURL', cu);
[flags, kv] = amt_configuration('silent');
[flags, kv] = amt_configuration('silent', 'normal');
```

Description:

`amt_configuration` accepts the following optional parameters:

- `'cacheURL', cu` - Set the download URL of the cache
- `'auxdatapath', ap` - Set the path where the auxdata is stored
- `'auxdataURL', au` - Set the download URL for the auxdata
AMT_CONFIGURATION - Get and set the configuration of the current AMT session

Program code:

```matlab
function [flags, kv] = amt_configuration(varargin)

% AMT_CONFIGURATION Get and set the configuration of the current AMT session
%
% Usage:
% [flags, kv] = amt_configuration;
% [flags, kv] = amt_configuration('cacheURL', cu);
% [flags, kv] = amt_configuration('silent');
% [flags, kv] = amt_configuration('silent', 'normal');
%
% AMT_CONFIGURATION accepts the following optional parameters:
% 'cacheURL', cu Set the download URL of the cache
```
Documentation

>> help amt_configuration

`amt_configuration` Get and set the configuration of the current AMT session

Usage:

```matlab
{flags, kv} = amt_configuration;
{flags, kv} = amt_configuration('cacheURL', 'cU);
{flags, kv} = amt_configuration('silent');
{flags, kv} = amt_configuration('silent', 'normal');
```

`amt_configuration` accepts the following optional parameters:

- `cacheURL`, `cU`: Set the download URL of the cache
- `auxdataPath`, `aP`: Set the path where the auxdata is stored
- `auxdataURL`, `aU`: Set the download URL for the auxdata

`amt_configuration` accepts the following flags:

- `cacheMode`: Sets the global cache mode, supported options
  - `global`
  - `normal`
  - `cached`
Documentation

- Website: http://amtoolbox.org
  - Compiled version: the website
  - Code version
- In-code documentation
  - Syntax: Similar to reStructuredText
  - Compiler: Based on mat2doc
  - Automatic creation of figures and results
  - Links to the publications
EXP_TABUCHI2016 - Results from Tabuchi et al. (2016)

Usage:

```r
data = exp_tabuchi2016(flag)
```

Description:

`exp_tabuchi2016(flag)` reproduces figures of the study from Tabuchi et al. (2016)

The following flags can be specified:

`'fig1'` Reproduces the lower panel of Figure 6

Examples:

To display Fig. 6 use:

```r
exp_tabuchi2016('fig1');
```

References:

EXP_TABUCHI2016 - Results from Tabuchi et al. (2016)

Program code:

```matlab
function varargout = exp_tabuchi2016(varargin)

% EXP_TABUCHI2016 Results from Tabuchi et al. (2016)
% Usage: data = exp_tabuchi2016(flag)
% EXP_TABUCHI2016(flag) reproduces figures of the study from
% Tabuchi et al. (2016).
% The following flags can be specified
% 'fig6' Reproduces the lower panel of Figure 6
% Examples:
% --------
% To display Fig.6 use :
% exp_tabuchi2016('fig6');
% See also: tabuchi2016
% References:
% H. Tabuchi, B. Loback, T. Mezzani, and P. Najdek. The role of
% compression in the simultaneous masker phase effect. The Journal of the
% Acoustical Society of America, 140(4), 2016.
% URL: http://amtoolbox.org/amt-1.5.8/doc/experiments/exp_tabuchi2016.php

% #Author: Hisaoji Tabuchi (2022)
% #Author: Clara Hollmay (2023): adaptations for AMT
% #Author: Joonas Guevara (2023): implemented plotting

definput.import = ['amt_cache'];
definput.flags.type = ('missingflag','fig6');

[flags,~] = lutarghelper([],definput, varargin);
```
Documentation

- Website: http://amtoolbox.org
  - Compiled version: the website
  - Code version
- In-code documentation
  - Based on mat2doc
  - Syntax: Similar to reStructuredText
  - Automatic creation of figures and results
  - Links to the publications
- Model list and ratings
THE AUDITORY MODELING TOOLBOX

Models included in the AMT (latest release, 1.5.0)

In order to describe the quality of the models available in the AMT, we rate the implementation of every model by considering its source code and documentation. We also rate the models in terms of their verification, i.e., we rate the results of the implementation versus the results shown in the corresponding publication. The comparison is done within the experiments implemented in the exp_ functions. In the best case, the experiments produce the same results as in the publication - up to some minor layout issues in the graphical representations.

The following table provides an overview of the available models, their documentation, code, and verification status.

<table>
<thead>
<tr>
<th>Peripheral models</th>
<th>Function</th>
<th>Doc</th>
<th>Code</th>
<th>Verification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gammatone filterbank</td>
<td>gammatone</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Linear filtering for monaural masking (basic)</td>
<td>dau1996</td>
<td></td>
<td>✓</td>
<td>?</td>
</tr>
<tr>
<td>Linear filtering for monaural masking (improved)</td>
<td>dau1997</td>
<td></td>
<td>✓</td>
<td>?</td>
</tr>
<tr>
<td>Invertible Gammatone filterbank</td>
<td>holmann2002</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Dual-resonance nonlinear filterbank (DRNL)</td>
<td>lopezpoveda2001</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Fast-acting compression (CARFAC) model</td>
<td>illum2011</td>
<td></td>
<td>✓</td>
<td>?</td>
</tr>
</tbody>
</table>
Documentation

- **Website:** [http://amtoolbox.org](http://amtoolbox.org)
  - Compiled version: the website
  - Code version
- **In-code documentation**
  - Based on mat2doc
  - Syntax: Similar to reStructuredText
  - Automatic creation of figures and results
  - Links to the publications
- **Model list and ratings**
- **Other resources**
Getting started

- The release package:
  - For **working** with the AMT only
  - Download the **full** release package from Sourceforge
- Starting the AMT:
  - Go to the AMT directory
  - Start the AMT: `amt_start`;
  - Compile the environments: `amt_mex`;
- Testing the environment:
  - Simple test: `demo_absolutethreshold`
  - SOFA/auxdata/internet test: `demo_baumgartner2014`
  - C-compiler test: `demo_zilany2014`
  - Python test: `demo_verhulst2012`
- Stopping the AMT: `amt_stop`;
- Cheat sheet: [http://amtoolbox.org](http://amtoolbox.org)