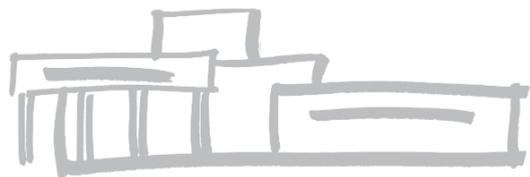


The Open Master Hearing Aid (openMHA)

4.18.0

Calibration Instructions



HörTech

Kompetenzzentrum für
Hörgeräte-Systemtechnik

The Open Master Hearing Aid (openMHA) – Calibration Instructions
HörTech gGmbH
Marie-Curie-Str. 2
D-26129 Oldenburg

LICENSE AGREEMENT

This file is part of the HörTech Open Master Hearing Aid (openMHA)

Copyright © 2005 2006 2007 2008 2009 2010 2012 2013 2014 2015 2016 HörTech gGmbH.

Copyright © 2017 2018 2019 2020 2021 HörTech gGmbH.

Copyright © 2021 2022 2023 2024 Hörzentrum Oldenburg gGmbH.

openMHA is free software: you can redistribute it and/or modify it under the terms of the GNU Affero General Public License as published by the Free Software Foundation, version 3 of the License.

openMHA is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Affero General Public License, version 3 for more details.

You should have received a copy of the GNU Affero General Public License, version 3 along with openMHA. If not, see <<http://www.gnu.org/licenses/>>.

Contents

- 1 Introduction** **1**
- 2 Calibrating custom hearing aid outputs** **1**
 - 2.1 Considerations regarding calibration of custom hardware used as hearing aid outputs 1
 - 2.2 Measuring the frequency response for a custom hearing aid output hardware with openMHA with an IEC 711 coupler 2
 - 2.3 Using the measured frequency response to calibrate a hearing aid output device 4
 - 2.4 Test your calibration 5

1 Introduction

The Hörtech Open Master Hearing Aid (openMHA) is a software platform for algorithm development and evaluation. For certain aspects of algorithm evaluation, correct calibration of the test setup may be a requirement. Since openMHA, being a software platform, does not include hardware (like hearing aid housings, microphones, and receivers), users of the openMHA are expected to supplement suitable hardware to pick up and produce sound at the wearers' ears. Users then face the challenge of how to correctly configure the software to calibrate the system.

This guide is a collection of best practices that the authors and users of openMHA employ to ensure their setups are calibrated correctly. We will extend this guide with new procedures as we add them to the openMHA, and as we receive user contributions in the field of calibration. openMHA users should be aware that calibration is an involved topic. Users need professional, specialized acoustic measurement equipment in order to be able to perform accurate measurements, and understand how to use their equipment. Understanding how to calibrate commercial hearing aids is a requirement for using this guide. The book "Hearing Aids" by Harvey Dillon contains a suitable treatment of the topic in chapter 4 of its second edition (2012). We cannot guarantee that the procedures presented here are free of errors. If you discover something that is wrong, please get in contact with the openMHA project, e.g. by filing an issue with our github project. We will do our best to improve the documentation and the tools that we provide as we go along.

2 Calibrating custom hearing aid outputs

This section describes how to measure the frequency response of a custom hearing aid output hardware, and how to use this frequency responses to calibrate the output of the device.

2.1 Considerations regarding calibration of custom hardware used as hearing aid outputs

Hearing aid receivers typically have a low impedance and a non-flat frequency response. When attaching hearing aid receivers to the output of a computer's sound card, care has to be taken to ensure that the sound card can actually drive the receiver, i.e. provide the necessary current without voltage drops. Otherwise serious non-linearities will result. Consider the application of a suitable amplifier to drive your output hardware.

A suitable acoustic coupler has to be used to connect the measurement microphone to the hearing aid receiver. The coupler will simulate the mechanical properties similar to a real ear canal. The coupler will also influence how the measurement microphone of your measurement equipment has to be calibrated. Please refer to Dillon and to the manuals of your measurement equipment for details.

When you calibrate custom hearing aid hardware, you always calibrate the complete output chain in combination, consisting of the individual sound card, any amplifier that you employ, and the hearing aid receiver.

2.2 Measuring the frequency response for a custom hearing aid output hardware with openMHA with an IEC 711 coupler

Hearing aids need to produce ear canal sound levels, not free field sound levels. For this reason, we suggest to also apply the Real-Ear-Unaided-Gain (REUG) in the openMHA output calibration. Please refer to Dillon for an explanation of REUG.

When calibrating hearing aid devices, you need to make a choice whether you are ok with calibrating the device for an average ear, or for the individual ear of a specific subject. In the following text, we will describe only the average calibration method. This may change in future.

Calibration requires reliable measurement of acoustic levels. This can only be achieved in a quiet environment. Please perform all acoustic measurements in a suitable, sound-proof booth or a similar environment. If your calibration equipment supports it, consider to filter out very low frequencies (e.g. < 20 Hz). If you notice fluctuations in the measured level, e.g. resulting from doors on other levels in your building, or from traffic outside, consider moving to a better isolated location, or to off-peak hours. To ensure that the influence of unavoidable noise is minimal, acoustic calibration is performed at relatively high sound levels around 80 dB. If you cannot achieve these levels with your custom output hardware, you should consider modifying the output hardware rather than measuring a lower sound level.

We will be using the MHA to produce test sounds of which we then measure the acoustic levels. When producing test sounds with high frequencies, take into account that the sound card will apply a low-pass filter for anti-aliasing that affects already frequencies below the Nyquist frequency. This can be an issue if you want to calibrate your system for relatively low acoustic sampling rates: If e.g. you use 16 kHz sampling rate in your sound card, measuring the sound level at 8 kHz should be avoided, and you should not reuse the measurement taken at 6 kHz for higher sound card sampling rates.

The calibration process will probably run into practical limits, and will require compromises. E.g., hearing aid receivers typically produce significantly softer sound levels for the same electric voltages at low frequencies. Exact calibration requires filters that invert the frequency response of the hearing aid output hardware, which would mean in this case, amplify with a high gain at low frequencies. However, a high enough gain at low frequencies will cause the output signal amplitude to exceed the dynamic range of the sound card. Since it is not possible to produce signal with a higher amplitude than the maximum amplitude of the sound card, the signal would need to be limited after the calibration filter is applied. This limiting will cause audible artifacts. To avoid these situations, it is usually a better choice to avoid too high amplification at the boundaries of the transmitted spectrum and to accept that the output levels are softer than intended by the hearing aid signal processing algorithms in these frequency areas.

2.2 Measuring the frequency response for a custom hearing aid output hardware with openMHA with an IEC 711 coupler

We describe here how to use a coupler according to IEC 60 711 to measure the frequency response of custom hearing aid output hardware. Compared to the 2cc coupler, the IEC 711 coupler has the advantage that for an average ear, the Real-Ear-to-Coupler-Difference (RECD) is small enough that it does not need to be taken into account when doing measurements to calibrate for the average ear.

Please assemble and calibrate your measurement equipment to perform measurements with the IEC 711 coupler. Please refer to the manual of the equipment manufacturer for instructions.

2.2 Measuring the frequency response for a custom hearing aid output hardware with openMHA with an IEC 711 coupler 3

Note that some calibrator devices require that you take correction factors into account when calibrating your equipment for the IEC 711.

When your equipment is calibrated, then you can measure the frequency response of the output hardware with the help of the openMHA:

- Install openMHA on the computer where the calibration is performed (see README.md).
- We will use GNU Octave/MATLAB tools to control the measurement. To prepare,
 - Start GNU Octave/MATLAB from the shell where you have set the openMHA environment variables.
 - Change to the directory `mha/tools/mfiles`.
- Start the jack sound server with the same sampling rate that you will later use with the openMHA.
- Connect the custom hearing aid output hardware to the desired sound card output channel and to the measurement equipment.
- Use the GNU Octave/MATLAB function `measure_fresponse` provided with openMHA to measure the frequency response:
 - Call this function with the following parameters:
 1. the name of the sound card channel in jack, e.g. `'system:playback_1'`,
 2. the sampling rate of the jack server in Hz.
 - The function returns a struct containing the fields:
 - Frequencies** A vector of frequencies in Hz of the test tones,
 - dBFSfor80dB** A vector of corresponding levels in dB re full scale (dB FS) of the sound card where the acoustic level was 80 dB.
 - REUGdB** A vector of corresponding suggested correction gains in dB to compensate for REUG.
 - correctionsdB** A vector of corresponding correction gains, initially all 0.
 - example: `response = measure_fresponse('system:playback_1', 48000);`
 - The function will start an MHA and begin to output test tones.
 - Your measurement equipment which is connected to the output will measure the acoustic level produced by each test tone.
 - Your goal is to alter the output levels so that acoustic levels of 80 dB are produced by the test tones:
 - If the acoustic level is different from 80 dB, then adjust the level by entering the required difference to reach 80 dB.
 - Repeat the corrections until your setup produces 80 dB acoustic output level.
 - When you have reached 80 dB, enter a correction value of 0 to move on to the next test tone.

You have now the frequency response of your setup in a struct in GNU Octave/MATLAB. You can use all the tools available in GNU Octave/MATLAB to inspect and alter the measured frequency response in this struct before using it to calibrate the MHA. We provide the GNU Octave/MATLAB function `plot_fresponse` that you can use to plot the frequency response.

The plot will display the frequency response of the required filter to calibrate the MHA for your output hardware. This filter inverts the frequency response of your output hardware, and adds the gains to apply the REUG.

You now have to decide if the overall dynamic exhibited by this filter is small enough that you want to apply the calibration filter as it is, fully compensating the averaged frequency response, or if you want to introduce correction gains to avoid too high gains at high or low frequencies, by changing the `correctionsdB` vector in the frequency response struct in GNU Octave/MATLAB.

2.3 Using the measured frequency response to calibrate a hearing aid output device

The openMHA plugin `transducers` is used to calibrate inputs and outputs. `transducers` can be configured with FIR filter parameters and a peak level to compensate for the frequency responses of hearing aid microphones and receivers. To make use of this capability, choose a openMHA configuration that already includes the `transducers` plugin, or create a new configuration that contains `transducers`.

When you are content with the calibration filter's frequency response, you can convert the frequency response struct to a set of filter coefficients and peak level that can be used in openMHA configuration. The GNU Octave/MATLAB function `fresponse_to_cfg` can be used to convert the frequency response struct to a text string that can be used in the openMHA configuration.

The function `fresponse_to_cfg` prints out values for the `peaklevel` and for the `fir` configuration variables of the openMHA plugin `transducers`.

You can take these values and insert them into your configuration file, into the `transducers` configuration. Remove the ellipses (`. . .`) printed, and remove superfluous semicolons. These are printed by the `fresponse_to_cfg` for cases where you have multiple output channels. You have to assign the peaklevels and FIR filter coefficients to the correct output channels that correspond to the measured hardware.

Additional GNU Octave/MATLAB tools that we want to mention here briefly:

1. For target devices with low computational power, using the FIR filter of plugin `transducers` (computed in the time domain) may be too computationally expensive. For openMHA configurations running on devices like these, an alternative may be to adjust the frequency response with an `equalize` plugin in the frequency domain. If you want to take this route, you may want to consider to use our GNU Octave/MATLAB function `fresponse_to_eq`. As always, please inspect this function before using it for calibration purposes.
2. For calibrating the portable hearing lab (PHL), you may want to consider our application note in openMHA [example 22](#).

2.4 Test your calibration

It is crucial to test your calibration. Please configure a simple openMHA with the calibration values that you have measured, insert the sine plugin into this openMHA, and play back sinusoids with different frequencies and different levels. A suitable configuration file is present in the `examples` directory of openMHA. Please use the configuration file in the `test-calibration` subdirectory and alter it to contain your specific calibration values. The configuration can then be started with `mha ?read:test-calibration.cfg cmd=start`. You can either modify the sinusoid frequencies in this configuration file and start openMHA repeatedly with different values of `mha.sine.f`, or use the network or GNU Octave/MATLAB based tools to modify this value in the running openMHA.

When playing back sinusoids at the measurement frequencies, check the output levels with your measurement equipment. It is now expected that the output levels for 80 dB sinusoids is measured as $80 \text{ dB} + \text{REUG}(f) + \text{correction}(f)$.

Differences of up to a few dB are expected. You should check that over the total spectrum, the differences are equally distributed to higher and lower output levels than expected.