Open Speech Platform: Quick Start Guide

http://openspeechplatform.ucsd.edu

Release 2020a
June 16, 2020

Abstract

This document describes download, build, install and test steps for the Open Speech Platform (OSP) Release 2020a software.

This work is supported by:

- NIH R33-DC015046: Self-fitting of Amplification: Methodology and Candidacy
- NIH R01-DC015436: A Real-time, Open, Portable, Extensible Speech Lab to University of California, San Diego.
- NSF IIS-1838830: Division of Information & Intelligent Systems, "A Framework for Optimizing Hearing Aids In Situ Based on Patient Feedback, Auditory Context, and Audiologist Input"
- The Qualcomm Institute

Please visit OSP Forum - Getting Started to report bugs and suggest enhancements.

Contents

Chapter 1

Release 2020a Installation

This section goes over what is required to install the Open Speech Platform software and what are the steps needed for the different installation methods.

The installation process of OSP may take around $30-90$ minutes, depending on your computer, download speeds over the internet, and any installation errors that you may encounter and resolve.

1.1 Requirements for OSP

1.1.1 Equipment Requirements

In order to use OSP, you must use either a Mac or a *debian-based* Linux machine (such as Ubuntu or Linaro) with the following processing, memory, and storage requirements. Figures 1.1 and 1.2 provide a reference for these requirements.

1. **Processor**: Equivalent to an Intel Core i5 processor.
2. **Memory/RAM**: At least 8GB or more.
3. **Free Storage Space**: At least 2GB or more.

Figure 1.1: System requirements for a Mac operating system computer.

Figure 1.2: System requirements for a Linux computer on Ubuntu.

To check if your computer meets these specifications...

- **On Mac**, click on the Apple menu icon at the top of your screen. In the dropdown menu, choose "About This Mac". The specifications should look similar to Figure 1.1. Apple has more information for finding computer specifications.
- **On Linux**, you may need to use a terminal that accepts command lines to figure out the specifications. Figure 1.2 is a reference of what the specifications look like on Ubuntu 18.04.1, though this may appear differently for different Linux systems. You check out this Stack Exchange post for answers related to Ubuntu: askUbuntu - How do I check system specifications?.

After the installation, to verify that the system can deliver audio output, **you need some way to input and output audio**. Ideally, a working device such as a headset or pair of headphones would be used, but your computer's built-in microphone and speakers are good enough as long as the volume settings are not set to mute.

1.1.2 Installation Requirements

Finally, these are the additional applications and tools needed to successfully install OSP.

- **Command Terminal**: You will need to know how to operate the command terminal with working knowledge of basic terminal commands and features. This is to navigate through different folders and operate OSP after installation. Fortunately, this guide will cover all of the commands needed, as long as you follow the steps in order.
- **GitHub and git**: Our files are stored online via our GitHub Page. You will need to install git within the terminal, if this hasn't been done already. You will also need to know the `git clone` command to retrieve the files through the terminal.

1.2 Download Files from OSP

1. **Open the terminal application.**

   (a) On Mac, you can go to "Applications" within your Finder and type in the search bar "terminal". It should appear as a thumbnail that looks like a black box.

   (b) On Linux, there are many methods to open the terminal. An article on How-To Geek's website covers this: Four Ways to get Instant Access to a Terminal in Linux.

   (c) In the terminal, enter the command `cd /`. This will navigate you to the root directory.

   (d) Try to enter the command `ls`, which shows a list of folders within the directory that you're in and help you assess whether you're in the correct directory.

   (e) For now, it is recommended that you navigate to the Downloads directory, which is where the installation files will be located. For Finder or File Manager in Linux, check where the "Downloads" directory is and see what order of the directory names are listed. Enter the command `cd (name1)/`, where (name1) is the directory that you want to navigate to (for example, you would enter `cd Downloads/`).

   (f) Repeatedly enter the `ls` and `cd (name1)/` commands as necessary until you are in the Downloads folder.

2. **Install git.** By now, you should be in the Downloads directory in your terminal. If you already have git installed in your computer, skip this step and proceed to step 3. If not, follow these steps:

   (a) Type in `sudo apt-get install git` to install git in your computer.

   (b) For Mac systems, you can install git from the following link: https://sourceforge.net/projects/git-osx-installer/files/.

3. **Download the 2020a Release from GitHub**
   On your browser, navigate to
   https://github.com/nihospr01/OpenSpeechPlatform-UCSD to download

the latest release. Press the "Clone or download" button in GitHub. You can either "clone the software on your computer" or "download ZIP file to your computer."

(a) **Use the `git clone` method.**
If you wish to clone, click on the button to the right of the text (it should look like a clipboard, see Figure 1.3). This would copy the line of code.

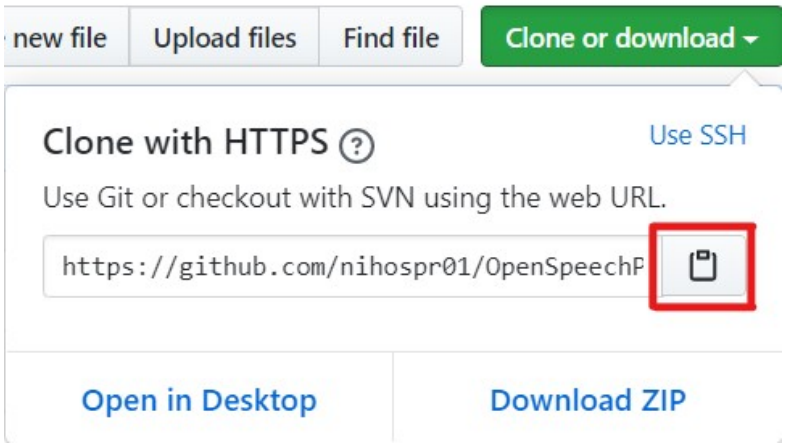

Figure 1.3: If you are using the `git clone` method, click on this button (outlined in red). This will copy the line of code, which you will paste into the terminal.

(b) Navigate to the Downloads directory using the commands `cd (name1)/` and `ls` commands outlined in Step 1, if you haven't already. Then Refer to Figure 1.4 for the expected output.

```
$ git clone https://github.com/nihospr01/OpenSpeechPlatform-UCSD.git
Cloning into 'OpenSpeechPlatform-UCSD'...
remote: Enumerating objects: 7909, done.
Receiving objects:   6% (535/7909), 238.29 MiB | 373.00 KiB/s
```

Figure 1.4: Expected output from the git clone command. Files for OSP should be downloading, which will take a considerable amount of time, depending on your internet download speed.

(c) **.zip file method** From the "Clone or download" button, click on "Download Zip" (you may refer back to Figure 1.3). You may need to wait a few moments before the browser prompts you to download the .zip file, which should have "OpenSpeechPlatform" attached to its name.

i. Mac computers support opening .zip files. Once you download the .zip file, go to the "Downloads" within Finder and open the .zip file. Move

ii. Debian-based Linux computers may need a third-party application (such as 7-Zip) to open the .zip file. Extract the folder within the .zip file and move it to an appropriate place within the file manager application.

4. After doing the git clone command or downloading and extracting the .zip file, **enter the command `cd OpenSpeechPlatform-UCSD/Software/Build-Scripts`, to navigate to a folder named "Build-Scripts" before proceeding to the next steps of the installation.** You will install OSP's software packages needed within this folder. Depending on your use, you will have multiple options and flexibility with regards to installation.

1.3 Choosing the Installation Method

- **Real Time Master Hearing Aid (RT-MHA)** - This is the hearing aid algorithm which takes the audio from the environment and modifies it for the listeners specific prescription.
- **Embedded Web Server (EWS)** - Is a process that makes any web browser enabled device into a graphical user interface which can control the RT-MHA algorithm. There are two flavors of EWS available in this release written in two different programming languages. The NodeJS version is currently being developed and will replace all of the functionality found in the PHP version.

Below are the available options to go about the installation process.

- **Installing Everything - RT-MHA and Node.js version of EWS** This is the latest version of OSP. Future releases will completely adopt Node.js.
- **Installing Everything - RT-MHA and PHP version of EWS** This is the legacy version of OSP, which will eventually not be used in future releases. This version still has the "Goldilocks" page.
- **Installing/Updating just the RT-MHA**
- **Installing/Updating just the Node.js version of EWS**
- **Installing/Updating just the PHP version of EWS**

Before you beginning any of these installation methods, here are additional disclaimers:

- In the terminal, **be sure that you have already navigated to the "Build-Scripts" directory**. Otherwise you will not be able to properly install the files. The file path to the "Build-Script" directory should be `OpenSpeechPlatform-UCSD/Software/Build-Scripts/`, as Step 4 of Section 1.2.
- **If you are using Ubuntu, you may have to manually install the latest version of a software package related to Node.js called "node-pre-gyp" as well as the "npm" package.** to resolve installation issues and error messages in advance. This step will eventually be fixed in a future release.

1.3.1 Installing Everything - RT-MHA and Node.js version of EWS

1. Run the command `./install_all_njs`, which does the following:
   (a) Identify the operating system (OS) on your computer – currently OS X, Debian and Redhat/Fedora based Linux (It will work with apt-get and yum package managers).

(b) Install all the pre-requisite software packages.

(c) Build and install RT-MHA as well as the Node.js version of EWS.

2. To check whether the system has successfully installed in your system:

   (a) In the current terminal, run `osp`, this should start running the OSP.

   (b) In a separate terminal type `ews-backend`. This should start running the backend of the Node.js version of EWS, this backend will also have the frontend built-in.

   (c) Open a Browser window and type the URL 0.0.0.0:5080. This will open the webpage to the NodeJS version of EWS.

1.3.2 Installing Everything - RT-MHA and PHP version of EWS

1. Run the command `./install_all_php`, which does the following:

   (a) Identify the operating system (OS) on your computer – currently OS X, Debian and Redhat/Fedora based Linux (it will work with apt-get and yum package managers)

   (b) Install all the pre-requisite software packages.

   (c) Build and install RT-MHA as well as the PHP version of EWS

   (d) Finally, it installs osp in `/usr/local/bin/osp` and a script to invoke ews in `/usr/local/bin/ews`.

2. To check whether the system has successfully installed in your system:

   (a) In the current terminal, run `osp`, this should start running the OSP.

   (b) In a separate terminal type `ews`. This should start running the PHP version of ews.

   (c) Open a Browser window and type the URL `0.0.0.0:8080`. This will open the webpage to the PHP version of EWS.

1.3.3 Installing/Updating just the RT-MHA

1. **This step is only needed if you are installing RT-MHA for the first time.** Run the command `./pre_req_all`, which command will identify your OS install all the necessary pre-requisite packages in your system

2. Run the command `./libosp`, which installs the librtmha and RT-MHA in your system.

3. To check whether the RT-MHA has successfully installed in your system, In the current terminal, run `osp`, this should start running the RT-MHA.

1.3.4 Installing/Updating just the Node.js version of EWS

1. **This step is only needed if you are installing this version of EWS for the first time.** Run the command `./pre_req_all`. This command will identify your OS install all the necessary pre-requisite packages in your system

2. Run the command `./ews_njs`. This will install the Node.js version of EWS on your system..

3. To check the EWS installation run `ews-backend` on the terminal, You can open a Browser window and type the URL `localhost:5000`. This will open the webpage to the Node.js version of EWS.

1.3.5 Installing/Updating just the PHP version of EWS

1. **This step is only needed if you are installing this version of EWS for the first time.** Run the command `./pre_req_all`. This command will identify your OS install all the necessary pre-requisite packages in your system

2. Run the command `./ews_php_public`. This will install the PHP version of EWS on your system.

3. To check the frontend run `ews` on the terminal, You can open a Browser window and type the URL `0.0.0.0:8080`. This will open the webpage to the PHP version of EWS.

Chapter 2

Release 2020a Package Testing and Validation

This chapter describes how to check that the installed software package(s) for OSP are working properly.

2.1 Connecting Your Audio Device

Note: For Mac computers, the default audio input and output built into the computer is sufficient for testing the audio. You do not have to following these steps to connect your audio device, but it is recommended to do so.

2.1.1 On Mac Computers

1. Connect your audio device to the audio or USB port on your computer.
2. **Open Audio MIDI Setup**: This can be found in `Finder | Applications | Utilities`. Figure 2.1 shows the settings for audio *input* and Figure 2.2 shows the settings for audio *output*.
    - For general audio headsets and headphones, set the format to 48,000 Hz. Levels should be 1.0.

- For any high end audio interface box (such as Zoom TAC-8), set the format to 48,000 Hz. (Additional step to guide the person to set the option to 24 bit.)

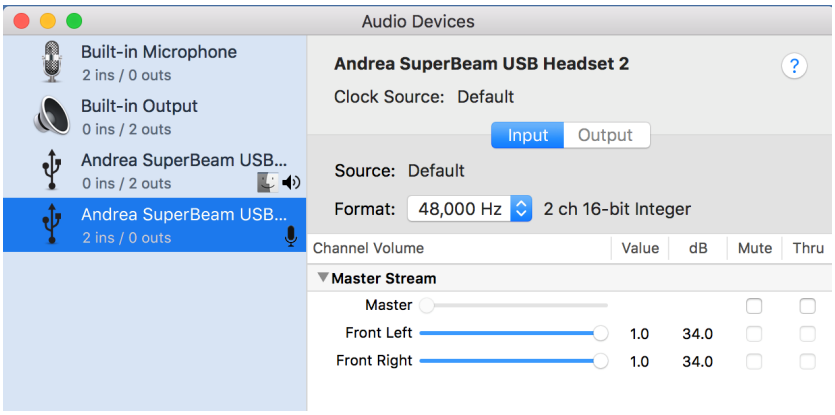

Figure 2.1: **Audio input settings** within a Mac computer. For most devices you should set the format to 48,000 Hz and the levels to 1.0.

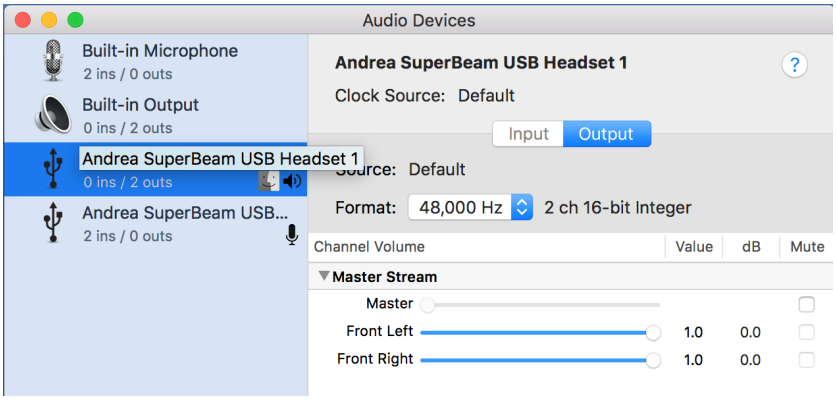

Figure 2.2: **Audio output settings** within a Mac computer. For most devices you should set the format to 48,000 Hz and the levels to 1.0.

2.1.2 On Linux Computers

1. On Linux machines, the default audio is not set. Within the terminal, enter the command `pa_devs` to give the list of the available devices and the device numbers they correspond to.

2. Scroll up in the terminal to view a list of different devices and single-digit numbers associated with each device. Run the command `osp --input_device x --output_device y`, where `x` is the number associated with the device that inputs sound and `y` is the number associated with the device that ouputs sound.

   Figure 2.3, for example, shows the command issued as `osp --input_device 6 --output_device 6`, where 6 is the value associated to the connected audio device.

```
Default low input latency   =  -1.0000
Default low output latency  =   0.0058
Default high input latency  =  -1.0000
Default high output latency =   0.0348
Default sample rate         = 44100.00
Supported standard sample rates
 for half-duplex 16 bit 8 channel output =
        32000.00, 44100.00, 48000.00, 88200.00,
        96000.00, 192000.00
------------------------------------ device #6
Name                        = Andrea SuperBeam USB Headset: Audio (hw:1,0)
Host API                    = ALSA
Max inputs = 0, Max outputs = 2
Default low input latency   =  -1.0000
Default low output latency  =   0.0087
Default high input latency  =  -1.0000
Default high output latency =   0.0348
Default sample rate         = 44100.00
Supported standard sample rates
 for half-duplex 16 bit 2 channel output =
        44100.00, 48000.00
------------------------------------ device #7
Name                        = sysdefault
Host API                    = ALSA
```

Figure 2.3: List of devices displayed by running the **pa_devs** command

2.2 Test RT-MHA

You can interact with RT-MHA from command line interface (CLI) to display and change the HA state. **Please make sure that the audio device is connected and that you can hear sound from the device. Otherwise, go back to section 2.1 to connect the device.**

1. Do not wear the audio device for now.
2. Open or navigate to your terminal and enter the command `osp`. The terminal should show results similar to Figure 2.4.

```
Garudadris-MacBook-Pro-6:~ hgarudadri$ osp
4 threads available
Done
Input device # 3.
    Name: Andrea SuperBeam USB Headset
      LL: 0.00458333 s
      HL: 0.0139167 s
Output device # 2.
   Name: Andrea SuperBeam USB Headset
     LL: 0.00335417 s
     HL: 0.0126875 s
Num channels = 2.
TCP Server created
```

Figure 2.4: Output of the terminal when you issue `osp` command.

3. To familiarize you with the initial commands that OSP has, please enter the following series of commands and steps in the terminal.
   (a) `-p` This command will print the *complete state* of RT-MHA. In the terminal messages above, notice that the gain on the left and right channels is -20 dB, to account for overall gain of RT-MHA.
   (b) `--gain -15` This command, with a value of -15, will set the system volume to -15 dB. By default, the gain is set to -20 dB.

By entering this command, you are making RT-MHA louder by 5 dB. The -15 value can be changed to a different value.

(c) Type in `-p` to see the changed gain values. Your terminal should look similar to Figure 2.5.

```
-p
{"*left":{"en_ha":1,"rear_mics":0,"gain":-
20.0,"g50":[0.0,0.0,0.0,0.0,0.0,0.0],"g80":[0.0,0.0,0.0,0.0,0.0,0.0],"knee_lo
w":[45.0,45.0,45.0,45.0,45.0,45.0],"knee_high":[120.0,120.0,120.0,120.0,120.0,
120.0],"attack":[5.0,5.0,5.0,5.0,5.0,5.0],"release":[20.0,20.0,20.0,20.0,20.0,
20.0],"mpo":120.0,"noise_estimation_type":0,"spectral_type":0,"spectral_subtra
ction":0.0,"afc":3,"afc_delay":150,"afc_mu":0.004999999888241291,"afc_rho":0.9
850000143051148,"afc_power_estimate":0.0},"*right":{"en_ha":1,"rear_mics":0,"g
ain":-
20.0,"g50":[0.0,0.0,0.0,0.0,0.0,0.0],"g80":[0.0,0.0,0.0,0.0,0.0,0.0],"knee_low
":[45.0,45.0,45.0,45.0,45.0,45.0],"knee_high":[120.0,120.0,120.0,120.0,120.0,1
20.0],"attack":[5.0,5.0,5.0,5.0,5.0,5.0],"release":[20.0,20.0,20.0,20.0,20.0,2
0.0],"mpo":120.0,"noise_estimation_type":0,"spectral_type":0,"spectral_subtrac
tion":0.0,"afc":3,"afc_delay":150,"afc_mu":0.004999999888241291,"afc_rho":0.98
50000143051148,"afc_power_estimate":0.0}}
Done
--gain -15
Done
-p
{"*left":{"en_ha":1,"rear_mics":0,"gain":-
15.0,"g50":[0.0,0.0,0.0,0.0,0.0,0.0],"g80":[0.0,0.0,0.0,0.0,0.0,0.0],"knee_low
":[45.0,45.0,45.0,45.0,45.0,45.0],"knee_high":[120.0,120.0,120.0,120.0,120.0,1
20.0],"attack":[5.0,5.0,5.0,5.0,5.0,5.0],"release":[20.0,20.0,20.0,20.0,20.0,2
0.0],"mpo":120.0,"noise_estimation_type":0,"spectral_type":0,"spectral_subtrac
tion":0.0,"afc":3,"afc_delay":150,"afc_mu":0.004999999888241291,"afc_rho":0.98
50000143051148,"afc_power_estimate":0.0},"*right":{"en_ha":1,"rear_mics":0,"ga
in":-
15.0,"g50":[0.0,0.0,0.0,0.0,0.0,0.0],"g80":[0.0,0.0,0.0,0.0,0.0,0.0],"knee_low
":[45.0,45.0,45.0,45.0,45.0,45.0],"knee_high":[120.0,120.0,120.0,120.0,120.0,1
20.0],"attack":[5.0,5.0,5.0,5.0,5.0,5.0],"release":[20.0,20.0,20.0,20.0,20.0,2
0.0],"mpo":120.0,"noise_estimation_type":0,"spectral_type":0,"spectral_subtrac
tion":0.0,"afc":3,"afc_delay":150,"afc_mu":0.004999999888241291,"afc_rho":0.98
50000143051148,"afc_power_estimate":0.0}}
Done
-q
Done
Garudadris-MacBook-Pro-6:~ hgarudadri$
```

Figure 2.5: Terminal messages that show the output of values before and after executing the `--gain -15` command. Before the command, gain is set to -20 dB by default, -15 dB after the command is executed. Remember that you can view these values by using the `-p` command, as shown.

(d) `-h` This command should help you experiment with RT-MHA by generating a list of commands that you can use within OSP, shown in Figure 2.6.

```
-h
Welcome to the Open Speech Platform
Usage:
  osp [OPTION...]

 Control Signals options:
      --samp_freq arg      Set the sampling frequency for the mic and
                           reciever (default: 48000)
      --input_device arg   Please indicate which device you want to use
for
                           input
      --output_device arg  Please indicate which device you want to use
for
                           output
      --multi_thread arg   Please indicate if you want OSP to run in
multiple
                           threads
  -q, --quit               Quit OSP
  -p, --print              Prints out the current user data structure
  -h, --help               Prints out the help
```

Figure 2.6: High level commands for **osp** using the command line interface (CLI).

4. **Before you proceed with the next step, be careful to not put the gain too high, otherwise your ears may start ringing. You may need to change the numerical value in the command to a lower value instead of -15.** Wear the audio device and make sure the audio output connection is stable. Enter the command `--gain -15` again and listen for external audio stimuli. This is how you know that RT-MHA is working.

5. The last command to enter is `-q`, which will quit OSP.

2.3 Test EWS - PHP version

1. For testing EWS, open two terminals side by side.

2. In the first terminal, type `osp`. In the second terminal, type `ews`.

3. Open a browser, such as Chrome. Enter `http://localhost:8000` in the search bar. You will see the landing page as shown in Figure 2.7.

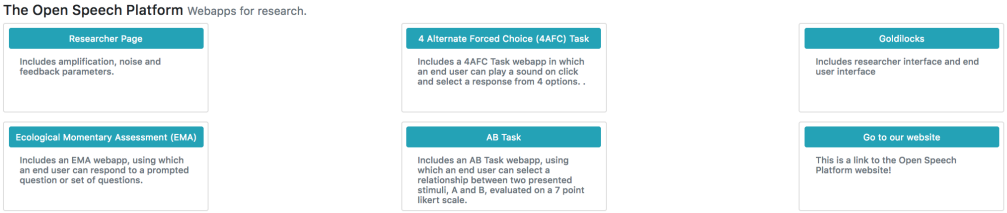

Figure 2.7: OSP Landing Page. Some of the apps are not yet connected to RT-MHA, but included here for early feedback on the user interface. These in progress web-apps are 4AFC and AB Task apps.

4. There are several blue-green buttons with labels. Navigate and click the one labeled "Researcher Page". What you should see is the Researcher Page interface, similar to Figure 2.8.

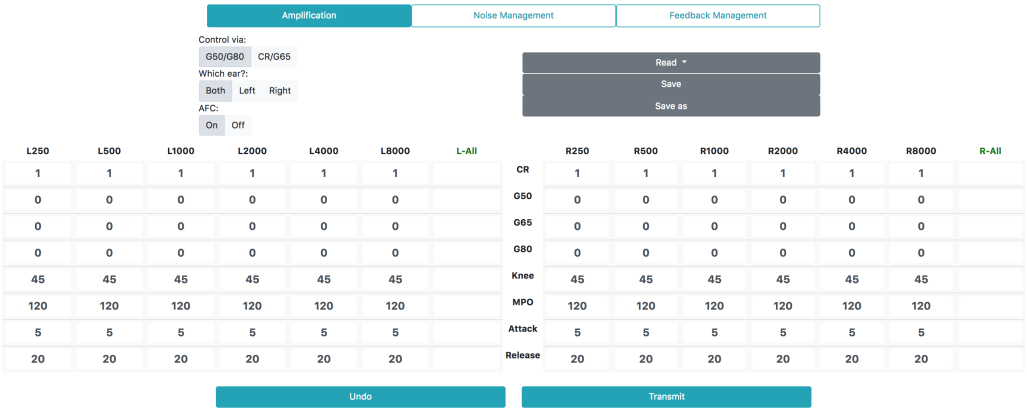

Figure 2.8: OSP Researcher Page. You can change Amplification, Noise Management and Feedback Management values from the first, second and third tabs, respectively.

5. You should be within the "Amplification" page. Pay attention to the settings area labeled "Control via", which have buttons to toggle either "G50/G80" or "CR/65". Choose "CR/G65".

6. You can now change gains in individual bands by entering numerical values for individual cells in the G65 row. In the G65/All cell, enter 5 for the value. You should see new values for RT-MHA highlighted.

7. Wear your connected audio device. Press the "Transmit" button below and listen. Your audio experience should be similar to when you entered the `--gain -15` command in the terminal.

8. Navigate to the terminal where you entered the `osp` command. Enter `-p`. You will notice values for the different settings for both the left and right audio channels: overall gain is -20, compression ratio (CR) is 1, and G65 is 5.

9. Navigate back to the Researcher Page on your browser. You should still be in the Amplification section. Navigate to the 3 cells corresponding to the G65 row and columns 250, 500, and 1000. Enter -15 for each cell. By setting this value for the 3 cells, you are setting a -15 dB attenuation for the 250, 500, and 1000 Hz frequencies.

10. Press the "Transmit" button below and listen. As a result of changing these values, you will notice that the low frequency noise is significantly reduced. Depending the headsets you are using, your experience might be different.

11. When you are finished, be sure to close both terminals to make sure you don't receive errors from the terminal from accidentally running OSP or EWS twice.

Figure 2.9: OSP Web Apps in Console Mode. If you are using a browser such as Chrome, you can right click in the browser window and choose Inspect. In this mode, you can view exchange of information between RT-MHA and the app. You can also change the app to be formatted for laptop and mobile devices using the Toggle Device Toolbar icon.

Chapter 3

Release 2020a Development - EWS Node.js version

Note - This chapter is still a work in progress and will be more complete in the next release.

This section is useful if you want to perform development on the Node.js version of the EWS. **Please make sure that you have all the necessary packages installed in your computer before moving forward with this section. Otherwise, go back to 1.3.2 or 1.3.5 to either install OSP and EWS - Node.js version or just EWS - Node.js version.**

Now you have the following options:

1. **Executing the Compiled Node.js Backend**
   Run the command `./ews_njs`. This will install the Node.js version of EWS on your system (which also contains the frontend) and will automatically execute the Node.js backend.

2. **Executing just the Node.js backend**
   Run the command `ews-backend`. This will execute the Node.js backend from the /user/local/bin directory where it has been installed

3. **Executing just the Node.js frontend**
   Run the command `ews-frontend`. This will execute the Node.js frontend from the local
   /osp-release-staging/Software directory.

4. **Executing the Node.js backend in development mode**
   Run the command `ews_njs_dev`. This will execute the Node.js backend in live development mode(using watch), any changes you make to the backend will be instantly reflected live in the webpages and recompiled to the backend(**Note: This code will not compile the frontend into the backend, for that you will need to run the script in step 1**)