EPFL

Design of an Externalized Music Player

Gloria **DAL SANTO**

Supervisor:
*Vincent **GRIMALDI***
Responsible:
*Prof. Dr. Hervé **LISSEK***

Spring Semester 2021

Ecole Polytechnique Fédérale de Lausanne – LTS2

Contents

1. Introduction

Introduction: Sound Externalization

- Sounds from the real-world are perceived **externalized** and the source can be **localized** → **Localization**

- Sounds from headphones are perceived **inside-the-head** → **Lateralization**

- Headphones **sound externalization** recreates the perception of an externalized sound source

- Applications in hearing aids technologies, augmented and virtual reality, …

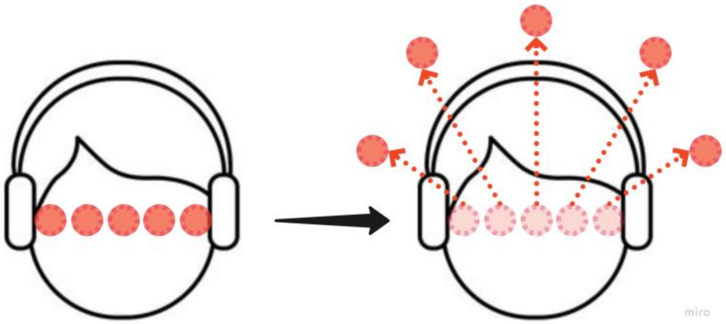


Figure 1: Concept of Sound Externalization

Introduction: Scope of the Project

- Sound reproduction systems over headphones may lead to **mental fatigue** which can be reduced with externalized sounds.

- The scope of the project is to design of a **music player capable of externalizing audio tracks** according to a specific environment
  - Based on **convolutional artificial reverberation**
  - Adjustable in real-time according to an **head tracking system**

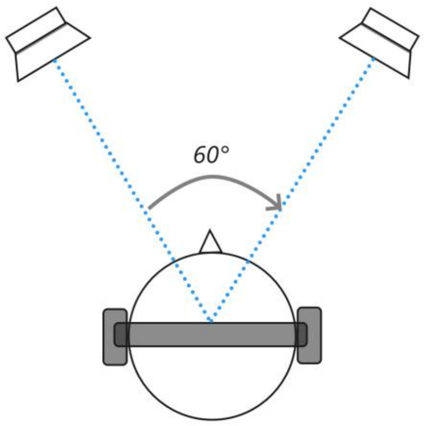


Figure 2: Simulated loudspeaker setup

Introduction: Convolutional Artificial Reverberation

- Artificial Reverberation Algorithms simulate the **psychoacoustic impact** of reverberation impulse response features
  - Input ‘dry’ **signal convolved with BRIR**
  - For real-time application, **partitioned convolution** is the most valid alternative to regular convolution
  - Partitioned convolutions guarantee a latency as short as the partition size (ex: 64 samples at 44100 Hz → 1.45 ms)

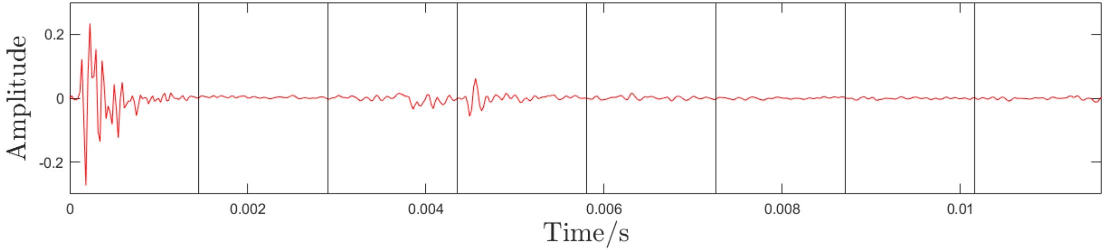


Figure 3: Impulse Response Partitioning

2. Method

Method: Tools

- Implementation on **Pure Data**
- Auxiliary scripts on **MATLAB** and **C** language
- BRIRs recorded at EPFL listening room using **KEMAR** manikin by **G.R.A.S.**
- HRTFs from the compact dataset made available by **MIT Media Lab**
- **IMU** sensor produced by **x-io Technologies**

Method: Pure Data

- **Open source** visual programming environment for interactive **computer music** and multimedia works
- Equipped with hundreds of objects optimized for real-time multimedia application
- Can interact with multiple devices (e.g. Raspberry board, Arduino board, MIDI devices) and allows transfer of data over WiFi

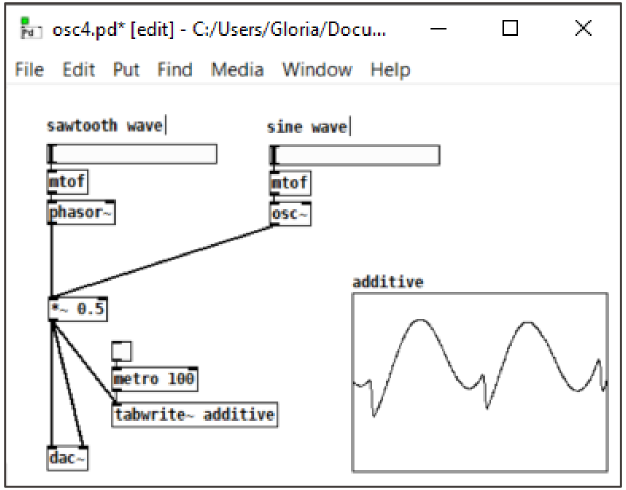


Figure 4: Example of a Pure Data patch

Gloria DAL SANTO

Method: Limitations

- No objects available for partitioned convolution that allow to change on the fly the set of BRIRs without producing artifacts

- The available objects for partitioned convolution
  - **Clear the buffers** when a new set of BRIRs are loaded
  - Works on a **heavy use of FFTs** that may cause spikes in the CPU usage which in turns interrupt the audio stream

- Lack of an intuitive tool for debugging custom objects and patches

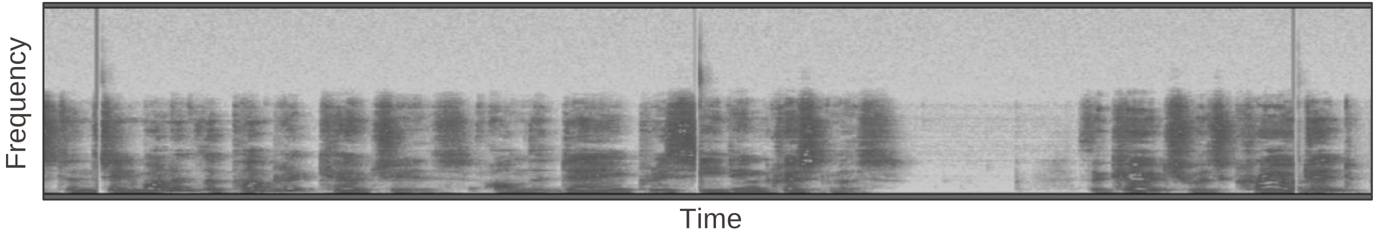


Figure 5: Spectrogram of the output of a convolution object in Pd

3. Concept

Concept: Initial Designs

Design 1

- Improve the source code of the available objects
  - Modified buffer management
  - Import the BRIR dataset at initialization

Conclusions

- The deadtime reduced slightly but not enough
- The code requires a more drastic re-design to get acceptable results

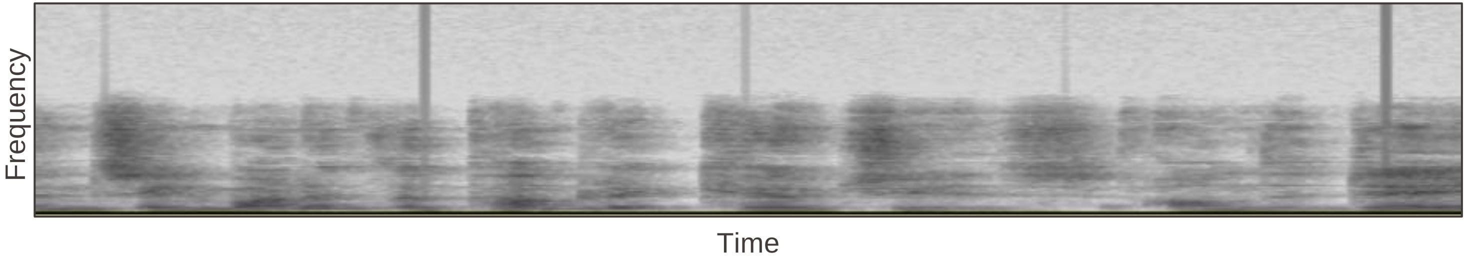


Figure 6: Spectrogram of the output of the modified convolution object in Pd

Concept: Initial Designs

Design 2

- Circular chain of 5 partitioned convolution objects per channel whose output is controlled by a multiplexer
  - The output of the multiplexer never coincides with the newly updated partitioned convolution object
  - Expected high usage of the CPU

Conclusions

- Same artifacts as in Design 1

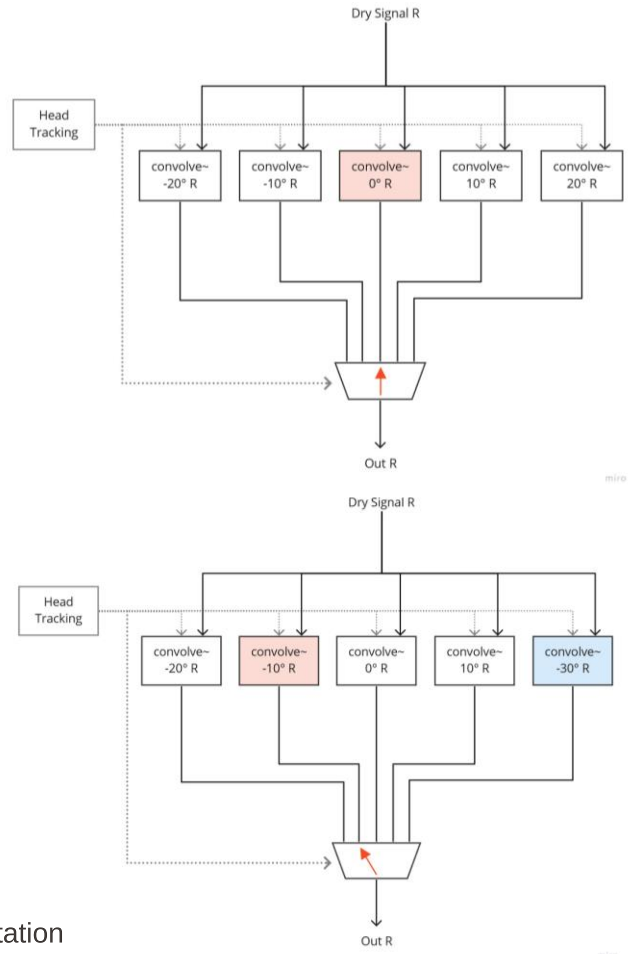


Figure 7: Two subsequent instances of [part of] the second implementation

Concept: Final Design

- Direct sound and reflections treated separately and combined together after processing
  - **Direct Sound:** dry sound convolved in time domain with the HRTFs
  - **Early Reflections** (ER) and **Late Reverberation** (LR)**:** dry sound convolved in frequency domain with modified BRIRs

- HRTFs are linearly interpolated
- The location of the ER and LR is fixed to avoid artifacts

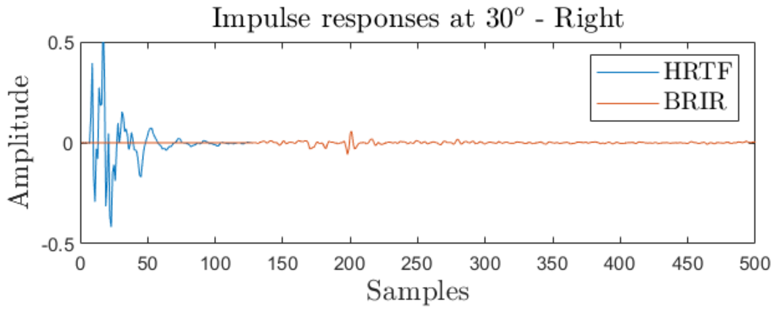


Figure 8: HRTF and modified BRIR at 30°

Concept: Final Design

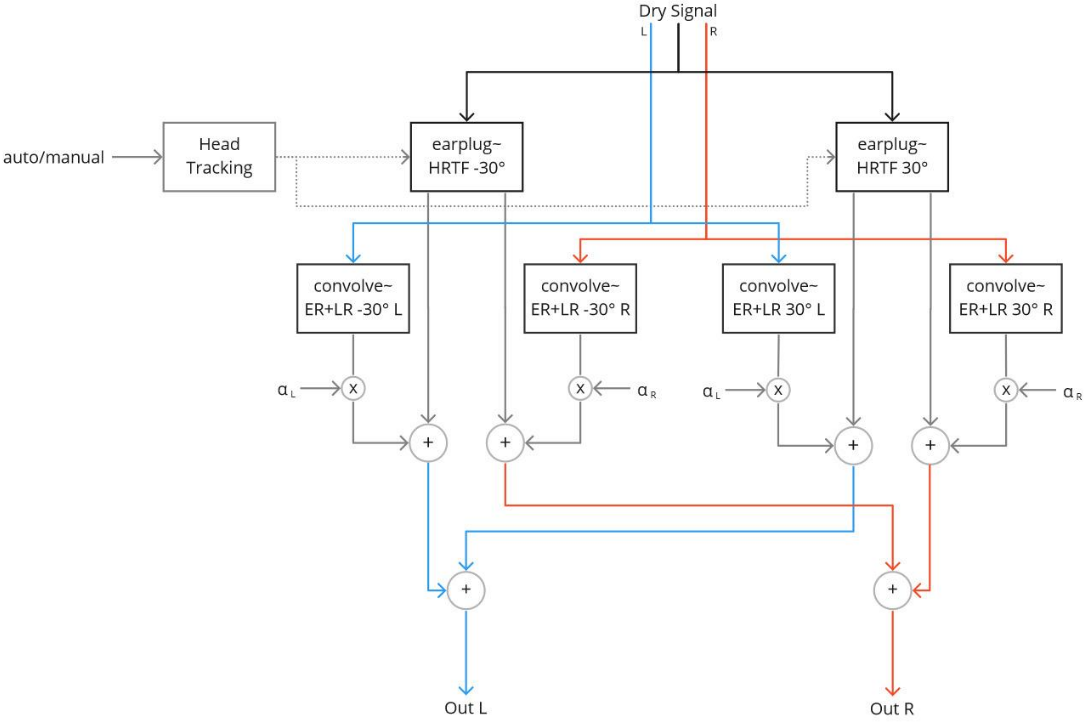


Figure 9: Diagram of the sound externalization block

Pure Data objects:

- **earplug:** convolution with the HRTFs
- **convolve:** convolution with the BRIRs

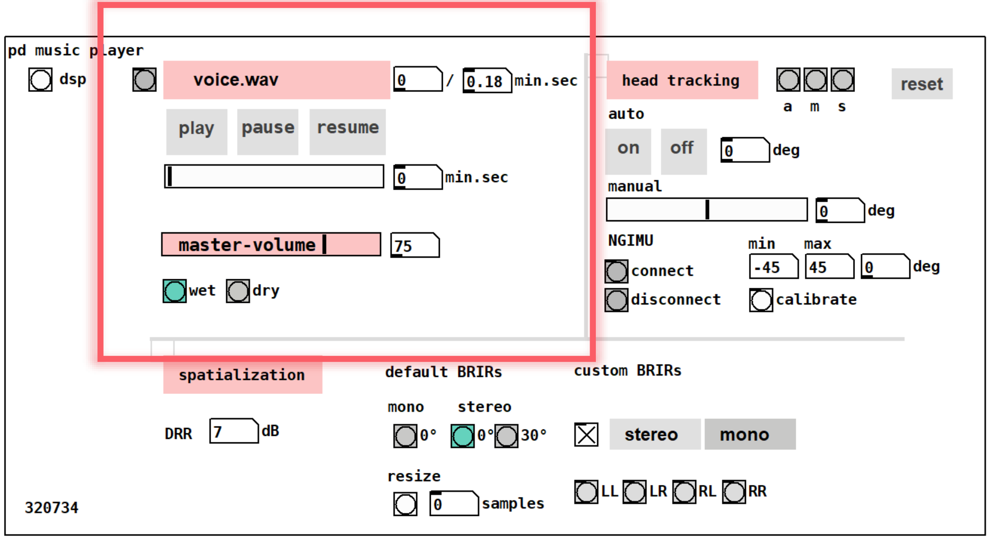
4. User Interface of the Music Player

Gloria DAL SANTO

Unser Interface

Music Player

- Load from file explorer
- Play/pause/resume the audio file
- Scroll bar to play from a specific point
- Master volume control bar
- Toggle between wet (externalized) and dry sound



Unser Interface

Head Tracking System

Three modes available

- **auto:** simulation of the rotation of the head from -45° to 45°
- **manual:** user can set the desired angle of the head
- **sensor:** connects the patch to the IMU tracking device to retrieve the rotation of the head

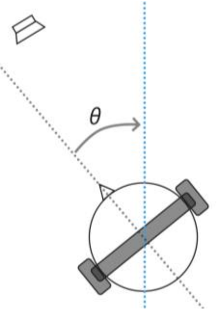
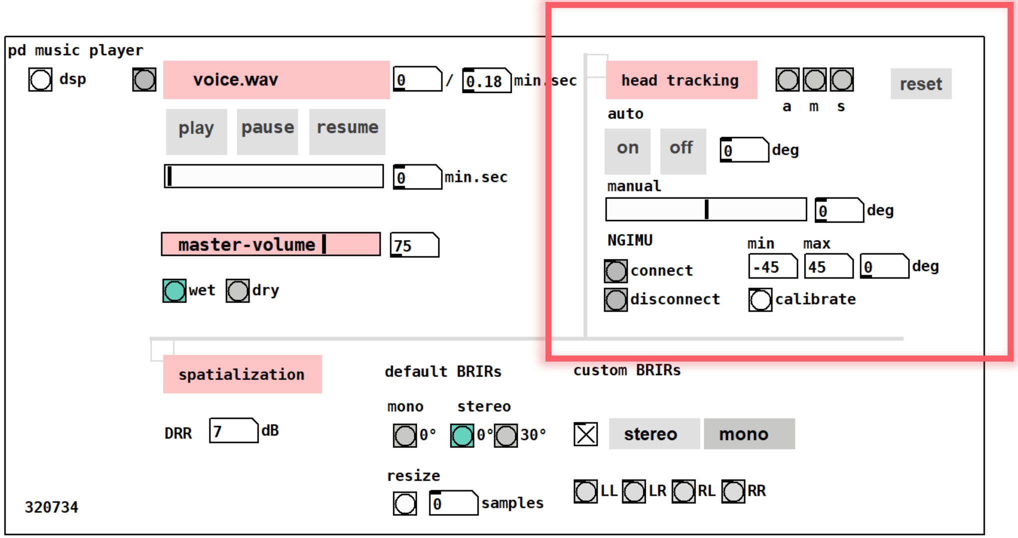


Figure 10: Head tracking system angle convention



Gloria DAL SANTO

Unser Interface

Artificial Reverb

- Custom **DRR**
- Simulation of mono or stereo loudspeaker set up
- Custom length of the BRIRs
- The user can load his own set of ER+LR both in mono and stereo mode

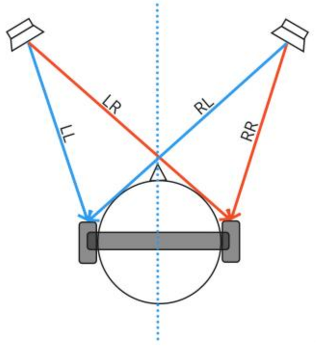
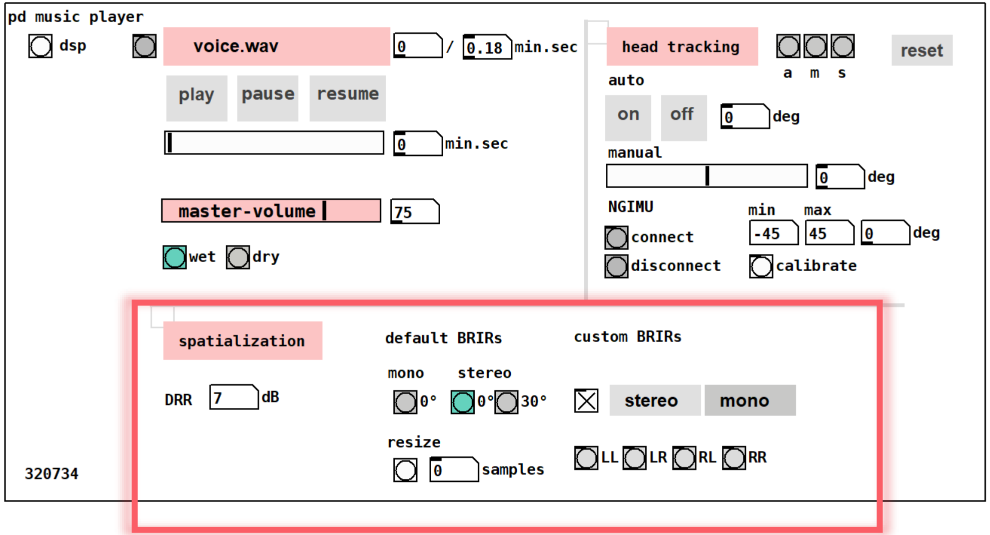


Figure 11: BRIRs naming convention



Gloria DAL SANTO

5. Conclusions

Conclusions

Strengths

- Control over several parameters
  - It can be used to investigate the effects on perceived externalization of different configuration
- Artifact-free and low computational expensive implementation
- Modular architecture, easy to be modified

Limitations

- It cannot be distributed as a standalone application
- Lack of a Pure Data object that allows to change the set of BRIRs in real-time

Future Work

- Conduct a formal psychoacoustic test to assess the efficacy of the sound externalization algorithm and on the effect of externalized sounds on mental fatigue

EPFL

Thank you for your attention!

•

Questions?

Gloria Dal Santo
gloria.dalsanto@epfl.ch