Assignment 3
EECS 4462 3.0 Digital Audio, Fall 2020, Section A

Due: Friday, November 13, 2020, 11:59pm.
Format: Individual.

Creating a digital reverb

The purpose of this assignment is to give you experience using the dsp module in JUCE, as well as understand impulse responses and convolution.

To get started

This assignment does not require any further software to be installed. Your setup from the previous assignments should be sufficient. Starter code and some useful links are posted under the Assignment 3 section on the course webpage.

Run the Projucer, and create an empty plugin project. Add the DSP module in the included modules as shown in the video. Export to your IDE, and replace the default code with the starter code.

What to do

For this assignment, you have to implement a digital reverb in four different ways:

1. Using the `dsp::Reverb` processor provided by JUCE. The GUI of your plugin must allow the user to set all the parameters that can be set through the API of this processor.

2. Using recorded impulse responses. Search the web for reverb impulse responses. There are plenty of free ones available. They are usually in `.wav` format.

   Your GUI must allow the user to select any of at least 5 different reverb types based on these impulse responses. Each type must described accurately on the GUI. The user must also be able to load their own impulse response.

   Place the impulse response files in a directory called `Resources` on the desktop. Your plugin must retrieve the impulse responses from there (see the slides for how to do this with JUCE). The evaluators of your plugin will be asked to place the `Resources` directory on their desktop.

3. Using constructed impulse responses. Construct an AudioBuffer that contains values that emulate an impulse response, i.e. they start close to 1 and decay exponentially to 0. Your GUI must present at least two such impulse responses (one with fast decay, and one with slow decay). It must also provide an option to add some randomness to each sample to achieve more natural results.

   The length of the AudioBuffer will determine how long the reverberation will last. Your GUI must allow the user to adjust this length anywhere from 0 to 6 sec.

4. Using multiple delays with feedback. Reverberation is essentially slightly delayed copies of the original signal that get weaker and weaker. These copies are usually also high-passed versions of the original signal as low frequencies are absorbed by most walls.

   Build a simulation reverb using multiple delays with feedback where the delayed copies are highpassed. The user should be able to adjust the delay times and the cutoff frequency. The maximum delay time does not need to be longer than 1 sec. Your plugin must choose default values for these parameters that present the best reverb you were able to construct using this method.

   You can use the delay implementation you prepared for A2, or use the Delay processor linked from the assignment page.

How to Submit

Click on Assignment 3 Submission in the course eClass page, and follow the instructions. The link will be available one week before the deadline.