Music and Technology 2 Project Blog
Alex Vaughan

1/9 - I made the document to log updates and note any progress I make for my Music and Technology 2 project.

1/17 - three research papers appropriate for my interests:
https://quod.lib.umich.edu/i/icmc/bbp2372.2018.023/--electronic-music-documentations-practices-and-models?view=image - study of electronic music documentation and a proposal of a documentation model to allow better recording and transmission of electronic music

https://quod.lib.umich.edu/i/icmc/bbp2372.2018.051/--digital-ensemble-skills-in-a-laptop-orchestra?view=image - a set of digital ensemble skills attributes for collaborative music making in a laptop orchestra

https://www.nime.org/proc/nime2023_46/index.html - How machine learning has been used and where it has been used in musical expression

1/21 - Final project ideation:
My goal for my final project is to create a digital music piece made in a DAW or similar program. I plan to possibly learn a new DAW or work with an existing one, and work to create an original piece. I also want to learn how to create my own custom plugin or something similar that can be imported to DAWs for my own use in order to leverage my programming experience. This will make my piece more unique.

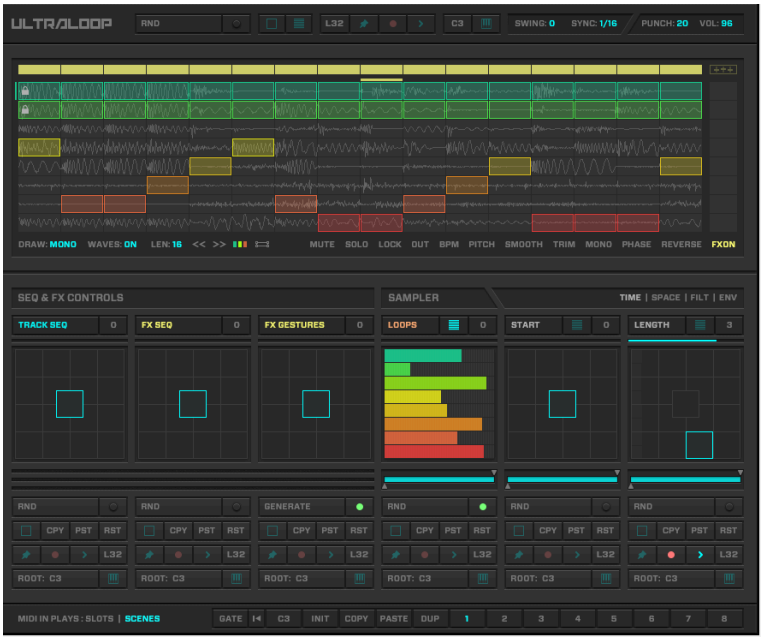
1/28
Have been making the first steps towards prototyping the plugin in Max. The plugin randomly selects different volume mappings and multiplies the mappings with the input source. In the future, I plan to generate a sequence of random numbers that dictate which mapping will be used, so the plugin is consistent over replays and not randomized every time.

2/7 - I found that Max actually supports converting Max patches directly into other formats, such as VST3 plugins, using RNBO - https://rnbo.cycling74.com/. I can also export my patches into C++ code for additional tweaking. The way RNBO does it is when you want to convert your patch, it is uploaded to a cloud compiler, then an online tool chain converts it to the plugin or format that you requested. I can use this to potentially skip rebuilding the program from the ground up. The only issue is that it is a subscription-based service costing $10 a month on top of the Max 8 subscription of $13 a month. I'm currently looking online for a way to get it for free. There is a way to get it for free on MacOS, but I don't have a macbook to do that on.

2/27 - I am back to working on the Max 9 patch after deciding to circumvent the payment process by making a new account each month for the free trial. I have made good progress on the patch. Now, the subpatch that acts as the plugin takes in any signal before chopping up the signal into notes which have a randomized amplitude envelope. The length of the notes is

configurable. There is also a random 10% variation in the length of each note. I've also added the ability to change the pitch of the signal in a controlled manner, and added a small chance to skip a note or to double the length of the note. Unfortunately, my computer cannot seem to keep up with Max, and the patch lags out my entire computer and causes stuttering while open.

3/12 - I've been messing around with plugins to see what makes a good plugin, and I've spent some amount of time with the plugin 'Ultraloop'. The plugin allows you to play different parts of sequences, smoothly interchanging between them. What was really notable to me when I started messing around with this is the sheer number of user customization options. There are 8 tracks on the sequencer, 12 sequence modifiers at the bottom of the sequencer, variable sequencer and effect controls, customizable options for the start, length, panning, volume, filter, envelope, attack, hold, decay, and release for each of the 8 tracks. You can also choose to randomize any part or parts of the plugin, as well as modify certain aspects to work at different times of the loop. I never realized exactly how much customizability is given to the user in a plugin.




3/17 - I have come up with a direction I want to continue developing my Max patch towards. For now, I plan to add some more effects and user customization. Once I add that, I'm going to add a feature to "seed" the patch in order to get reproducible and saveable randomization. Basically, I'm going to add the option to add some amount of data into the patch, and every time that data is put in, the random elements of the patch will behave the same way every time. I will accomplish this by removing the expr(random) elements and replacing them with a new, manual formula for random number generation, one that allows for seeding. Then, I'll add the option to insert a seed, or to just pick a random one generated with the expr(random) function. This way, if the random elements end up sounding good, the seed can be saved, so when running the patch again, it will sound the same.

3/24 - I have implemented the custom random number generation, and it works. Now, you can input a seed number and get a set effect from the patch.

3/31 - I am rewriting my core patch to be more organized and to allow for greater user modification. I already have the main component of the patch working, which is the chopping up of input signal. I am in the process of implementing the random variance in pulse lengths, which

will also allow for the simple toggling of the variance as well as the ability to control how much variance there is. After I implement that, I plan to implement the pitch shifting, with user control over whether it is on or not, the strength of the shift, and the duration of the shift.

4/8 - I implemented random note skipping. It is toggled on or off, and has a dial that determines the frequency that notes are skipped at. I've also added an inlet to allow users to change the seed of the Random patch. Now, I just need to implement the pitch shifting. I also have a few ideas for increasing the scope of the project - perhaps an option to store seeds and play them back at an interval to create rhythm? It seems very feasible.

4/18 - final presentation: M&T2 Presentation