Lluis Suros
24 de enero de 2019

Lab 2

Drone generation with customised additive synthesis

For this task I got inspired by the Additive Synthesis example provided in the students material and I used that structure as a starting point. I found it a good case for learning how subpatches work in Pd.

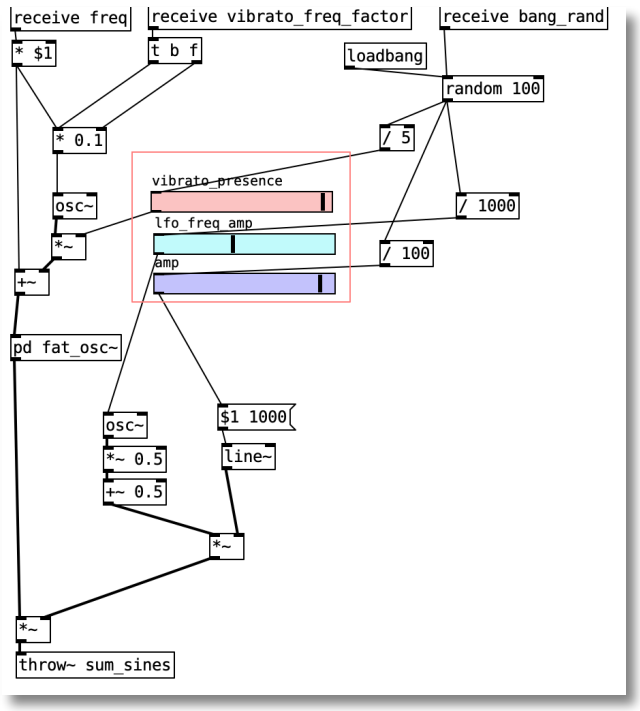
The question that I wanted to explore is how would it sound, if each harmonic in additive synthesis would appear and disappear periodically, thus altering the timbre structure. Maybe something similar to the Indian Shruti Box, but also providing a GUI where the user can fine-tune each harmonic at will.

I wanted to use only Vanilla Pd so I could focus on learning the basics of the language instead of trying out different libraries and objects. I also wanted to create an instrument which could be used easily, and that could keep playing in an interesting on its own.

The final result contains a main file (AdditiveDrone.pd), and two files containing MainControl.pd and Harmonic.pd

I will try to discuss my thinking below.

Harmonic.pd

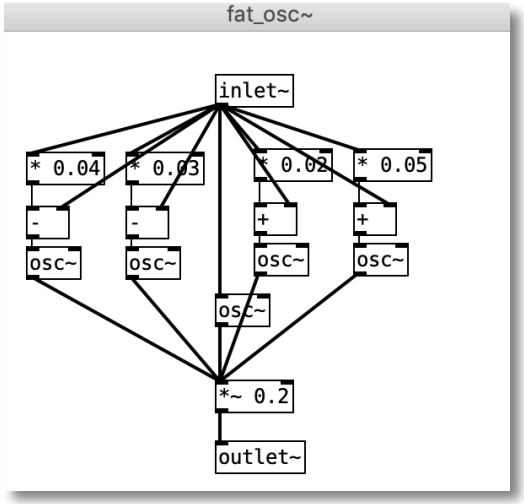


I added a vibrato to the harmonic generator, and also the planned amp LFO to allow the harmonic fade in and out. Note that I had to scale the amp LFO oscillator to make it range between 0 and 1.

After some trials I decided that I didn't want the user to access the individual vibrato frequency, and that I wanted it to be related to a main GUI control, otherwise it was becoming hard to use. With controlling the individual vibrato amount I think is enough. That is the reason why all harmonics receive a vibrato frequency factor (for example a 0.1) and they will multiply it by the own frequency to make the effect consistent over each overtone.

I also created a [fat_osc~] object which consists in 5 oscillators with slightly different frequencies, in order to get a "fat" sound (attached picture).

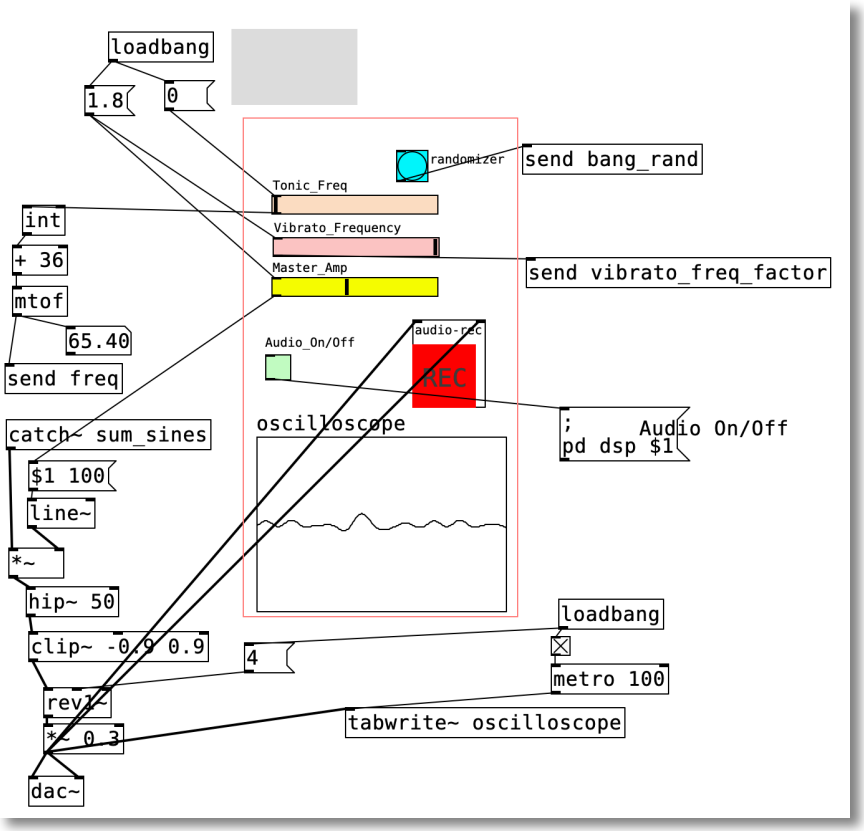
It is also here the code responding to each randomisation.



MainControl.pd

In this subpatch I send and receive the general information for all harmonic.pd. It is meant to be the main GUI for the user.

I allowed only one octave to be selected by the user, so it always sounds similar while keeping the possibility to adjust the tuning to other external sounds.



I also include the mentioned Vibrato frequency control, that every harmonic will receive and multiply by their own frequency, so each harmonic produces an equal relative amount of vibrato to their note.

Since editing the harmonics can be quite boring and slow, I also added a feature for randomising the values, so like this we can explore different setups (although I reckon it sound quite similar all the time).

I also spend some effort fine tuning the Master Amp part, because I wanted it to offer a little bit of distortion, and also some reverb afterwards, so I had to be careful that the values don't exceed over 1 after after the clipping part.

I designed this part so the user only needs to open the main patch and start making sound straight away. For fine tuning we can still edit each harmonic in a separate way.