

Computational Creativity Lab 3

Alia Morsi: u163640

January 29, 2019

1 Usage

1.1 Patch Summary

Just to get started, to listen to output, open system.pd. Then, click the toggle button on the markov patch, on the first main_patch_additive wav_gen box, and slide the amplitude in that same wave_gen box until you can hear a sound.

1.1.1 markov:

Implements first order Markov.

- **Note Duration** can be controlled via the tempo slider: This ranges from 200 to 400, and is used to set the metro object that times the writing of the next note, and is also used to set the duration of the midi note output. Since the duration of the note is the same as the time to wait between successive notes, there would be no silence.
 - However, only the frequency (and not the amplitude) is sent from the patches to my synth, so technically the 0 amplitude message to signal the stop of the note will not be processed as a stopping note.
- The boxes are from 0 - 11 and indicate the midi pitch
- Since the pitches are from 0 - 11 only, the octave number box can be used to raise the output by **octaves**. The possible range is 1 - 8 and defaults to 4.
- **To start the patch, hit the toggle box.**

To move between markov probability tables, there is feedback within the markov patch (indicated by the control connection that spans bottom to top in the patch, kept on purpose to be super clear that there is feedback). To see this in the markov.pd patch, observe the 2nd output of the coll object, where it is clear how the index is changing.

1.1.2 midi_2_freq_ampl:

Can be used for midi control (if the inlets are ignored and the GUI is used), or for midi display (if the inlets are used, the GUI responds to their input).

- The octave number can be modified from this box (if there is no input from the inlet), or it can be used to display what's coming in as input
- The pitch is displayed in the bang button keyboard and in the symbol boxes. Note that we only use natural and sharp symbols to indicate pitch.

1.1.3 main_patch_additive:

The additive synthesizer patch takes four input frequencies. Only one of them will take input from the markov patch. Only one other input is connected to a midi_2_freq_ampl patch, and the rest are kept disconnected, although they can be controlled via their sliders.

1.2 Limitations

1. Note that the accidentals (natural and sharp) should be more sophisticated if it were to take into account scale context to decide whether a note is sharp or flat (A flat not equal G# etc), but that was not considered for this assignment, although scale context is important if this shall could be used musically.
2. Musically speaking, the probabilities of first or second order would vary depending on the scale we are in. For example, If I am in C major, the probability of notes that follow an A is different than those that would follow an A if I were in A Major. However, this is understood as a limitation but not explored at the moment.
3. There is still no ADSR on the synthesizer. That was missing from last week and I did not get the chance to improve it.
4. Another problem is that any amplitude change with the wave_gen connected to the Markov generator creates a clicking sound. That's a huge problem but I'm not sure what should be done.
5. I guess it is important to carefully make a .coll file because at this point, there is not much aesthetic difference between a random note generator and my current system.

1.3 Ideas

It would be nice to have probability tables for each scale and to just change the scale using midi input (i.e from midi input you can get the tonic of the scale). That can still be Markov order 1 as we would consider only the past note, but would need a different representation (to view the probabilities as distances from

tonic, and then after calculating the necessary distance, add it to the current tonic).

2 Implementation

2.1 Overview of Choices:

- For this assignment, the synthesizer from last week was used (`main_patch_additive.pd`). Minor tweaks were made to it in terms of initialization and updating the inlets.
- The aforementioned `markov.pd` and `midi_2_freq_ampl.pd` patches are this week's additions to the system.
- Minor changes to some of the patches provided in the lectures. The ones that were used are provided in `lec_3_patches` folder of the submission
- `makenote` is used in the `markov.pd`, such that the output of the patch is
- the probabilities used in the collection can be found in `probabilities/lab3.coll`