

Design Document

Koso Suzuki

Bass Line Generator

This program will take into account the given parameters from the player using touchOSC to create a bass line using two Markov Chains. One Markov Chain will be used to generate a rhythm sequence, the other a pitch sequence. The probabilities for each state will depend on the previous state and the parameters provided by the player.

Parameters:

volume [0, 128)

output volume

density [0, 100)

density of notes- higher the density, lower the amount of rests in measure

complexity [0, 100)

complexity of notes- changes rhythm, pitches used

range [0, 100)

pitch range for the notes to be used

constantBass true or false

if true, disregards density, range, rootDownbeat, oneOctave and nonHarmonicTones and outputs constant bass with fixed pitch and duration determined by complexity, for each note

rootDownbeat true or false

if true, outputs root of the chord on each downbeat

oneOctave true or false

if true, use only one octave for the notes to be used

nonHarmonicTones true or false

if true, use non-harmonic tones

Features/Exceptions

-If density is 0, nothing is played.

-If complexity < 20, note plays on downbeat, duration determined by density.

-If $19 < \text{complexity} < 40$, note plays on downbeat, duration determined by density. If density > 74, another note is played during the middle of the measure. If meter is 3/4, the second note is on beat 3, if meter is 5/4, second note is on beat 4.

-If complexity > 39, notes are played at random times. Raising density will cause less notes to be rests. Raising complexity will cause smaller subdivisions of rhythm. If oneOctave is false and range is big enough, raising complexity will cause more leaps between chord tones.

-If range is 0, all notes that are not rests will be the note which is 3 octaves above the root of the chord.

-Changing range from 0 to 3 will allow the system to use the note a half step above the note which is 3 octaves above the root of the chord. ex) if cur_chord is C major, available notes are C(pitch 36) and C#(pitch 37). Adding 3 more to range will allow the note below the root, adding another 3 more will allow the note a whole step above the root, so on.

-If nonHarmonicTones is true, it will use diatonic non-harmonic tones if $59 < \text{complexity} < 80$, and both diatonic and chromatic non-harmonic tones for complexity > 79. All non-harmonic tones will be used as passing or neighboring tones, unless some notes are out of range, which the system will change its octave or use the root of the chord instead.