CSC 466: Candidate Research/Programming Project #1

Title: User-Oriented Algorithmic Composition Using A Constraint System & Genetic Algorithm

Algorithmic composition explores how music can be generated systematically using algorithmic approaches. Symbolic approaches lend themselves nicely to incorporating music theory fundamentals and understanding how pleasing music is composed.

I have been researching algorithmic composition techniques for my honors thesis project. A lot of the systems I looked at, while interesting, often left the human out of most of the music generation processes, with most of the human intervention being the human who coded the program. While this is a fine process for the programmer, there is a missed opportunity to increase music creation to a greater audience of people with the help of technology. This is where genetic algorithms and "interactive selection" come into play.

This project seeks to employ a constraint-based system and a genetic algorithm featuring interactive selection to create a generative music application where the user gets to interact with the music as it is being created. The constraint-based system will employ music theory to speed up the music creation process by limiting the solution space and aiming to create more musically pleasing results off the bat as opposed to random calamity. The constraint-based system will be used to generate the initial population for the genetic algorithm. The fitness metric is the user, who ranks each component of the music sample – melody, alternate melody, and bassline – out of ten, with zero being the worst and ten being the best.

This idea comes from two different sources: a video about an interactive melody maker using genetic algorithms and a scholarly journal proposing an interactive music generator using dynamic Markov chains and genetic algorithms. The genetic algorithm for this project is based heavily on the genetic algorithm in the video, while the hybridization of algorithmic composition

methods and the approach to generating multiple instrumental parts comes from the scholarly journal.

Music Representation

An individual in the population consists of the following fields:

- ***Melody*** – Melody consists of a list of pitches of a size determined by the user. It is the central element of the music sample.
- ***Alternate Melody –*** This melody is a melody that compliments the primary melody. The alternate melody may be a variation on the primary melody, a harmonization, or a whole new melody. Like the melody, it is simply a list of pitches of a size determined by the user.
- ***Bassline –*** Bassline is also complementary to the melody. In this project, the bassline will consist of notes of slower durations and lower octaves to differentiate from the melodies. The bassline also consists of a list of pitches of a size determined by the user.
- ***Rank –*** The rank is the user's rating of an individual (out of ten). The default rating is zero.
- ***Number*** – The number represents the position of the individual in a population.

The size of the music pieces will be limited by the number of measures a user specifies. This project is constrained to 4/4 time.

Music Theory Constraints

- This project is constrained to 4/4 time.
- All generated music will be constrained to notes from a predetermined key and corresponding scale.
- Melody note durations are constrained to quarter notes, eighth notes, sixteenth notes, along with accompanying rests of the same durations.
- Bassline note durations are constrained to whole notes, half notes, and quarter notes.
- Melody will be in the second and third octaves of EasyABC notation.

- Bassline will be in the first octave of EasyABC notation.
- The alternate melody will either be: (1) a permutation of the main melody (2) a harmonization of the main melody or (3) a completely new melody
    - The harmonization will be subject to music theory rules found here

Interactive Selection & Fitness

Unlike the RGB project, there is no hard-coded, numeric fitness metric. Instead, the user's preference is the fitness metric. For selection, a small number of random individuals will be chosen from the population for judgment. Each music sample will be presented one-by-one in a format such that the user can copy and paste the music segment into EasyABC for easy listening. After listening, the user enters one rating (out of ten) for the melody/alternate melody pairing and another rating (out of ten) for the bassline. This occurs for each sample in the interactive selection process.

Once all samples have been listened to and rated, the two top-rated individuals for melody/alternate melody and the two top-rated individuals for bassline are combined to make two parents. Essentially, parent #1 gets the top-rated melody/alternate melody and the top-rated bassline from the top-rated individuals, and parent #2 gets the second-place versions of those components. The remaining melody/alternate melody and basslines are copied and may or may not mutate depending on the mutation probability.

Mutation & Crossover

Mutation and crossover take the same form as the RGB project, except instead of altering one part of the Music object, alterations occur with all three parts – melody, alternate melody, and bassline. For mutation, each will have one pitch change to a random one, as defined by the constraint-based system. For crossover, the first parent's half of music – half of the melody, alternate melody, and bassline – comes together with the second parent's half of music to create a new individual in the population.

Analysis

Tweaks to the many variables of the genetic algorithm, as well as the constraint-based system afford many opportunities for analysis of music composition.