Reid Swanson
ISE 575
Week 9

The Use of Constraint Systems for Musical Composition

In this paper Wiggins shows how he uses Constraint Logic Programming to assist in the composition process and discusses why CLP is a natural solution for the task he is trying to solve. To help introduce his approach he first gives a brief introduction to Genetic Algorithms, how they have been used for musical composition and why they can also be considered a constraint based approach. After presenting his solution with CLP he concludes by raising some issues arising in musical representation that have not been well addressed which make computer generated music even more difficult than it should be.

Genetic algorithms are based on Darwin's theory of evolution and use two main operations to produce new generations. The first is “mutate” in which a bit or series of bits are a randomly changed. The other is “crossover” in which some genetic material (a sequence of bits) is swapped from one gene to another. A new generation is produced by applying these operations over the existing population. After the new population is spawned each individual is evaluated on by a fitness function which determines whether that gene will survive to produce a new generation. In this way GAs can be considered constraint based because generation is a potential solution that must satisfy the constraint of the fitness function. Although GAs are usually general purpose solvers domain specific biases in the mutations can be used and have been shown to improve their effectiveness.

This type of GA algorithm was used to produce a four-part harmonization where each chromosome was represented as a 4xn matrix of notes. The population was initialized randomly and a set of directed mutations were applied. The evaluation function was based on several rules of thumb used to produce harmonies. They found that the results were good enough to pass a first year undergraduate music composition class when submitted anonymously. However, after some analysis they discovered that any change in one voice was so influential to the result of every other that to reach a stable state many changes had to be made in a single step. However GAs are not good at this type of alteration and so no amount of directed mutations or number of generations could produce a stable state. Their conclusion was that it would probably be more beneficial to work on one voice at a time, starting with the cadences first, then bass lines and finally the inner parts, much like a human composer would do.

Serial compositions were introduced by Schoenberg as response to the trend music had been taking during the previous two centuries, in which more an more notes were being included, that were not in the same key as the piece. There is some theoretical discussion, based on physical properties of waves, why notes of the same key sound good together. This has to do with the ratio of the frequencies of the notes. Given a note, the closer the ratio, of the frequency of that note to the tonic of that key, is to one the less stable and more discordant the sound is. Schoenberg rejected the idea of a tonal center however and suggested a new formal way of composing pieces call Serial Composition. The idea was to compose using each of the 12 notes of the chromatic scale in equal numbers, and even suggested writing the 12 note sequences prior to composing the piece.

The focus of this work is to use CLP to follow Schoenberg's method and create a 12 note sequence that when played with four instruments would form a sequence of his 12 favorite chords. Each of the four instruments was designated to use a traditional form of transformation. The Flute would play the original series, the Oboe a retrograde, the Cello the original rotated four places and the Harp a retrograde series rotated 8 places. The search space to find a 12 note series that would satisfy the constraint specified above is 12! which is too large for a human or computer to exhaustively search for. However, using SICStus Prolog it was relatively easy for the author to specify a constraint satisfaction problem that was tractable. The only problem was that while recursing down the four lists the notes need to be kept in sorted pitch order, but since they weren't known in advance this posed a problem. However, a built in predicate “element/3” allowed for a solution to this problem. As it turned out several solutions returned and the author picked one that he particularly liked that harmonized with itself, yielding his favorite chords in more ways that were originally required by his constraints.

The final point of the paper raises an issue that is a complication for computational musical representation. It was discovered in the 17^{th} century that an instrument could be tuned to play in any desired key. However, this posed a problem because the frequency ratios of the notes in a scale would not add up to 2 in one octave as they were supposed to. A partial solution to the problem was to tune every semitone equally, called equal temperament, which allowed the ratios to sum correctly. However, not all instruments can be tuned this way, in particular vocals, or instruments based on physical structures. Perception also plays a role and can cause incorrect pitch assignment regardless. Although no solution was given how to remedy these issues it was suggested that a system is need that can work well on both equally and non equally tempered scales, which seems again to reinforce the idea of multiple representations.