Initial Report

Robin Harry Stephen Hawkins

February 3, 2014

Student no.: 1125589
Project Title: "Massively Collaborative Composition of Music by Evolution"
Supervisor: Dr. Kirill Sidorov
Moderator: Ralph Martin
Module: CM3203
Credits: 40

1 Description

Musical compositions are usually written by one person, or a small group of individuals. It is very difficult to involve large numbers of people in the writing process. For a large group of individuals to compose music together, each individual will only be able to make a small contribution to the whole. Non-musicians are not able to contribute to a composition with complex musical information. The input of all composers in such a scenario will have to be a series of simple, incremental refinements. This can be realised using an evolutionary algorithm.

Starting with a population of pseudo-randomly generated strings of notes (which may have very little inherent musical value), human participants in this project (composers) will rank the melodies against one another, serving as the fitness function. As the algorithm progresses and the generations advance, hopefully the average musicality of the population will rise. By the conclusion of the project, the individual melodies will be highly musical. Clusters of these melodies can then be proposed to the users/composers as a final composition. The users will be able to interact with the system via a web interface, which will allow them to listen to and compare melodies from the population. I will also be recording whether or not the users are musicians, for further analysis of the evolutionary process.

Autonomous evolutionary music composition has been done before, and even the technique of using humans in lieu of a fitness function is not without precedent: the GP-Music System ([1]) can use a either a human or a trained neural network as the fitness function. However, a crowd-sourced, massively collaborative approach to evolutionary music composition does not appear to have been undertaken. The ubiquity of the internet and multimedia-capable devices today makes projects on this scale possible. This may be the first time the musical ideas and preferences of a large group have been distilled into a single work.

2 Aims and Objectives

- Assess previous experiments in evolutionary music
- Develop suitable components for the evolutionary algorithm framework, namely:
  - Representation of melodies (what parts of the structure are considered atomic could change how fast the algorithm progresses)
  - Generation of initial population (this may not be as simple as random numbers)
  - Crossover/mutation (what techniques to use, how and when to use them)
  - Selection of individuals for crossover, when to initiate crossover
  - Replacement of individuals
  - Determine fitness rankings via human participants
  - Determine a stopping condition
- Construct a web interface for human participation
- Devise a method of converting between genotype and phenotype representations of melodies, ie., from string representation to sound.
- Crowd-source! Advertise and promote the project to drive traffic and gain participants
- Humans could be a bottleneck in this process devise a way of minimising or negating this effect
- Construct a back-end environment suitable for advancing the algorithm, storing a record of changes, and analysing the population and it's development.
- Design a suitable database schema to store the melodies.
- Analyse the change in the population between generations, identify patterns and successful genetic schemata

2.1 Outcome

The primary outcome of this project will be a composition with hundreds or thousands of composers, each contributing in a small way to the final piece. Additionally, I hope to determine, through analysis of generational change and the final outcome, what humans perceive as musical. Autonomous approaches to evolutionary music composition work to rules which are derived from music theory (for example, the Evolutionary Music Composer [2]) or some other heuristic. Although this approach is effective at assessing what is musically valid, it does not capture what people find attractive about music. By repeatedly polling large numbers of people for their opinion, I will be gathering data about what humans identify as musical. I can look back through the different generations for consistencies between highly successful melodies and through various analytical techniques, further derive what constitutes a good melody in the opinions of real people. As I will be restricting the rhythm of the melodies to a fixed-length (probably four bars) series of crotchets, the melodies' representation will be amenable to analysis as a vector or as a 1-d signal.

2.2 Predictions

People with a musical background tend to be far more accepting of (or even welcoming of) novelty in music — that is to say, a lack of repetition. I expect my results to confirm this — that people who do not self-identify as musicians will select more for repetition, and musicians will select for more complex or subtle features, like key signatures.

2.3 Other considerations

Insufficient participants: if too few people choose to participate within the time frame, it will be difficult or impossible to advance the algorithm to an extent that serious musical development is shown. The other possibility is that I will not get a broad range of perspectives, but rather just those of a few users. The latter possibility is preferable as it is non-catastrophic. The various social media sites offer some redundancy in this respect, as they allow me to reach out to different pools of individuals. It may also be possible to recruit other students to participate.

Trolling: the internet is not synonymous with courtesy and respect, so there

is some possibility that people will attempt to deliberately derail this project.

Technical failure: it will be impossible to test my system against high volumes of traffic until I have high volumes of traffic. As such, it will be hard to predict the effects. If there is a problem, it will have to be detected, diagnosed and remedied very quickly.

These are issues which are difficult to plan or test for. Some ingenuity will be required to prepare against them. There may be examples of other projects which have faced and surmounted these problems, which I can learn from.

3 Work Plan

Already done: I have a server and a programming environment set up and ready to go. Some exploratory coding has been done.

Weekly: meet with supervisor on Wednesdays. Ongoing: document progress.

Week 1

- Complete Initial plan.
- Construct basic website for user participation.
- Research evolutionary algorithm techniques
- Research other evolutionary music projects

Week 2

- Finalise aspects of evolutionary algorithm, eg. representation, crossover/mutation..
- Implement basic algorithm.
- Write code to play string representations as melodies
- Manually run the algorithm to benchmark the number of generations and rough number of steps before something musical appears

Week 3

- Build a back-end for analysis and testing (ie. Database. It may be possible to incorporate this into the algorithm code)
- Refine and test website
- Refine and test algorithm/back end

Week 4

- Launch Website
- Publicise and drive traffic
  - Canvas social media sites
  - Contact fan-base from music career
  - Ask other students to participate

Week 5

- Monitor progress (ongoing from now)

Week 6

- Identify and analyse popular melodies (ongoing)

Week 7

- Buffer week, allowing time for the unexpected.

Week 8

- Analyse final population (this could be in the following week)
- Analyse popular melodies for common features

Week 9

- Write report (ongoing)

Week 10

- Continue writing report.

Week 11

- Hand in final report.

The rate of progress may depend heavily on the number of users I can encourage to participate. These are guidelines only, and I do not expect them to be canonical.

References

[1] Music Generation With and Brad Johanson. Gp-music: An interactive genetic programming system for. 1998.

[2] Gustavo Abreu Yaser M. A. Khalifa, Hunter Shi. Evolutionary music composer. Submitted to Genetic and Evolutionary Computation Conference 2004.