Autonomous Evolutionary Music Composer

Y. Khalifa
Computer Engineering Department
State University of New York
New Paltz, NY 12561, USA
khalifay@newpaltz.edu

M. Basel Al-Mourad
Computer Science
Aston University
Birmingham, B4 7ET, UK
m.b.al-mourad@aston.ac.uk

ABSTRACT
A second-generation autonomous music composition tool is developed using Genetic Algorithms. The composition is conducted in two Stages. The first Stage generates and identifies musically sound patterns (motifs). In the second Stage, methods to combine different generated motifs and their transpositions are applied. These combinations are evaluated and as a result, musically fit phrases are generated. Four musical phrases are generated at the end of each program run. The generated music pieces will be translated into Guido Music Notation (GMN) and have alternate representation in Musical Instrument Digital Interface (MIDI). The Autonomous Evolutionary Music Composer (AEMC) was able to create *interesting* pieces of music that were both innovative and musically sound.

Categories and Subject Descriptors
J.5 Computer Applications: Arts and Humanities, Music

General Terms
Algorithms, Performance, Design, Human Factors, Languages.

Keywords: Music

1. INTRODUCTION
In [1], Gartland-Johnes and Colpey distinguish between two important objectives of search algorithms; exploration and optimization. Search algorithms, such as Genetic and Evolutionary algorithms, in creative applications, definitely serve the former objective. Most evolutionary systems in music composition are either interactive, where they need a tutor or a human evaluator, or autonomous, that posses automatic fitness assessment. An excellent review of the application of Genetic Algorithms (GA) in musical composition is provided in [2]. However, as stated in [3], most evolutionary composition systems listed in literature need a tutor, or a human evaluator in an interactive GA environment. The development of autonomous unsupervised music composers that possess automatic fitness assessment is still limited.

2. GA IMPLEMENTATION
The composition of music is performed in two Stages. In Stage I, a set of motifs is generated. In Stage II, motifs and their transpositions are combined to form two music phrases, A and B. At the end of Stage II, phrase $A^{\#}$ is generated by sharing each note of the phrase. At the end, a combination of $ABA^{\#}A$ is produced, which is one of the common combinations in music composition theory.

3. STAGE I
In Stage I, motifs are generated. A table of the 16 best motifs is constructed that is used in Stage II. These motifs will be used both in their current, and transposed locations to generate musical phrases in Stage II. Figure 1 shows the chromosome structure in Stage I. Each chromosome will contain 16 genes, allowing a maximum of 16 notes per motif. Each motif is limited to a four-quarter-note duration.

At the end of Stage I, a table of the top 16 motifs is constructed Each row in this look-up table represents a motif. The columns represent the different notes in the motif. Although all motifs generated are one whole note in duration, they could be composed of either one, two, four, six, or eight notes. However, single note motifs are highly discouraged.

3.1 Stage I Evaluation Function
In Stage I, only an Intervals Evaluation Function was used. Within a melody line there are acceptable and unacceptable jumps between notes. Any jump between two successive notes can be measured as a positive or negative slope. Certain slopes are acceptable, others are not. The following types of slopes are adopted: *Step*: a difference of 1 or 2 half steps. This is an acceptable transition. *Skip*: a difference of 3 or 4 half steps. This is an acceptable transition. *Acceptable Leap*: a difference of 5, 6, or 7 half steps. This transition must be resolved properly with a third note. *Unacceptable Leap*: a difference greater than 7 half steps. This is unacceptable.

If a leap is acceptable and resolves properly, no penalty will be assigned. There is also a possibility of bonus within the interval section. Certain resolutions between notes are pleasant to hear. We can define these bonus resolutions as the 12-to-13 and the 6-to-5 resolutions. The first is a stronger resolution, and therefore receives a larger weight. Thus the bonuses are calculated as in equations (1) and (2).

4. STAGE II
In stage II, two evaluation functions are implemented; Intervals, and ratio. The Interval evaluation function described in the previous section is used to evaluate interval relationships between connecting notes among motifs. Other evaluation function is described below.

GECCO'06, July 8–12, 2006, Seattle, Washington, USA.
ACM 1-59593-186-4/06/0007.

Ratios Evaluation Function. The basic idea for the ratios section of the fitness function is that a good melody contains a specific ideal ratio of notes, and any deviation from that ideal results in a penalty. There are three categories of notes; the tonal centers that make up the chords within a key, the color notes which are the remaining notes within a key, and chromatic notes which are all notes outside a key. Each type of note is given a different weight based on how much a deviation in that portion of the ratio would affect sound quality. The arbitrary ratios sough were: Tonal Centers make up 60% of the melody; Color Notes make up 35% of the melody; and Chromatic Notes make up 5% of the melody. Although these ratios choices could be quite controversial, they were a starting point and current ongoing work is looking further into making these ratios chosen by the user or music style dependent.

5. RESULTS

The four motifs in Figure 1 (a) to (d) were picked from the final 16 motifs chosen by the program. It can be observed that each motif has an identical rhythm consisting of four eighth-notes, one quarter-note, and two more eighth notes.

Examining motif *a*, the first three notes are all $F^{\#}$'s, indicating that no penalty will be assigned (a step size of 0). The next note is a $G^{\#}$, (2 half-steps away from $F^{\#}$). This transition is classified as a step and no penalty is assigned. The following notes are $F^{\#}$, $G^{\#}$, and E (a difference of 2, 2, and 3 half steps, respectively). These transitions are also acceptable.

Since each of the motifs in Figure 1 (a) to (d) contained an identical rhythm, it is of no surprise that a piece composed from these motifs contain the same rhythm. What is interesting to note, however, are the measures marked as I, II, III, and IV.

Measure I and III are the only measures throughout the entire excerpt in which the last two eighth-notes are not $G^{\#}$ and E. Measures II and IV are the only ones in which the first three eighth-notes are not all $F^{\#}$'s. The last note of measure I and the first note of measure II are the same. This is the result of the intervals evaluation function, since it's role in Stage II is to evaluate the transitions between motifs.

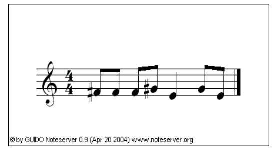
REFERENCES

[1] Gartland-Jones, A. Copley, P.: What Aspects of Musical Creativity are Sympathetic to Evolutionary Modeling, Contemporary Music Review Special Issue: Evolutionary Models of Music, Vol. 22, No. 3, 2003, pages 43-55.

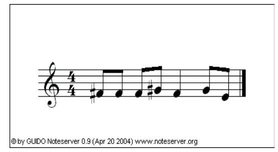
[2] Burton, A.R. and Vladimirova, T.: Generation of Musical Sequences with Genetic Techniques, Computer Music Journal, Vol. 23, No. 4, 1999, pp 59-73.

[3] Miranda, E.R.: At the Crossroads of Evolutionary Computation and Music: Self-Programming Synthesizers, Swarm Orchestra and Origins of Melody, Evolutionary Computation, Vol. 12, No. 2, 2004, pp. 137-158.

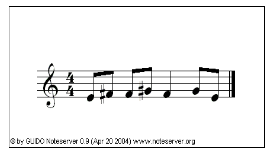
[4] Miranda, E.R., Composing Music Using Computers, Focal Press, Oxford, UK, 2001.



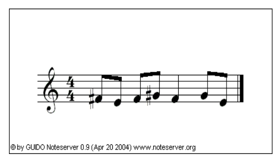
(a)



(b)



(c)



(d)

(e)

Fig. 3. (a) - (d) Sample motifs generated in Stage I of the Evolutionary Music Composer.

(e) A partial piece composed from motifs in the same generation as those in (a) through (d)