Melody Similarity and Tempo Diversity as Evolutionary Factors for Music Variations by Genetic Algorithms

Fernando H. Calderon Alvarado
Social Networks and
Human Centered Computing
National Tsing Hua University
Hsinchu City, 30013 Taiwan
fhcalderon87@gmail.com

Wan-Hsuan Lee
Institute of Information
Systems and Applications
National Tsing Hua University
Hsinchu City, 30013 Taiwan
l48711499@gmail.com

Yen-Hao Huang
Institute of Information
Systems and Applications
National Tsing Hua University
Hsinchu City, 30013 Taiwan
yenhao0218@gmail.com

Yi-Shin Chen
Institute of Information
Systems and Applications
National Tsing Hua University
Hsinchu City, 30013 Taiwan
yishin@gmail.com

Abstract

Music variations consists of modifying an original music piece so that it remains harmonious and provides a sense of novelty while still remain relatable to the original song. The melody should progress in a harmonious way and under a reasonable chord hierarchy, due to this constraints, music variations can be considered as a rule based composition system. In this research, we develop a genetic algorithm to compose a variation on a given music piece. To achieve this we design a novel evaluation function and evolutionary operators which favor the modification process. These are designed to include components of melody similarity and tempo diversity in addition to harmony. The experimental results show that the proposed system can generate variations that preserve musicality. A human evaluation study is also included to validate the proposed evaluation function. Additionally a link to listen to our generated compositions is provided.

Introduction

Music composition has traditionally been another mean of human expression. This process involves a combination of concepts from diverse domains including mathematics, music theory and creativity. Developments in the field of computer music have seen growing efforts in replicating such procedure. Automatic music composition is the process of generating music with the least amount of human intervention. In practice, it can be used to evaluate the degree to which computers can execute the task or as a tool to enhance the human process. This has led to a variety of implementations of music composition systems. Among this efforts we find music variations which are important to study due to their nature of creating over an existing structure. This behavior can provide insights on how the creative process can leverage on prior information and perhaps give us clues on how to apply these concepts in other fields.

The automatic music generation problem is complex and challenging. Recent studies propose music generation systems using neural networks and deep learning techniques and have achieved considerable success (Sturm et al. 2016; Chu, Urtasun, and Fidler 2016; Brunner et al. 2017; Yang, Chou, and Yang 2017; Yu et al. 2017). Another paradigm within the field focuses on composing music in a specific style instead of composing from scratch. Previous studies have focused on the generation of Chinese folk music, Jazz solos and fusions of Flamenco with Argentine Tango (Biles and others ; Liu and Ting 2017; Zheng et al. 2017; Luo et al. 2020). Moreover, algorithms have been able to generate specific style harmony based on a given melody. Providing a sense that composition systems can build on top of existing musical structures. Most of these techniques have focused exclusively on harmony instead of a global musical structure as its driving force.

The mentioned studies rely on a target style to generate or adjust a composition. Music variation refers to the process of modifying a musical piece without a target style. The lack of data with original and variation pairs limits the training and modeling by neural networks. Rule-based genetic algorithms do not require training data and have previously achieved good results in music composition and variation (Özcan and Erçal 2007; Majumder and Smith 2018; Alfonseca, Cebrián Ramos, and Ortega 2006). Music variation needs to consider other aspects as compared to traditional composition. In particular, they have to pay attention to the original piece and carefully select modifications that can build on it. This progression must still remain pleasing to a human audience. This requires balancing the amount of diversity added to the variations so that they remain novel while the original songs are still perceivable. In most of the previous studies the evaluation functions driving the evolutionary process rely solely on harmony rules or music theory. We propose a more complete function which considers the following aspects:

- Harmony: Evaluates if the result follows the classical music theory which favor musicality.
- Diversity: Measures the amount of rhythm variation added to the original song while being musically pleasing.
- Similarity: Determines how close is the modified melody to that of the original song.

These measurements can direct the evolutionary process and also be used as evaluation functions for any given pair of music compositions. In conclusion, the contribution of this research is to provide an automatic music variation system which relies on a new evaluation method which considers if the result is harmonious as well as the novelty factor in the

Proceedings of the 11th International Conference on Computational Creativity (ICCC'20)
ISBN: 978-989-54160-2-8

composition while still remaining identifiable to its source.

Methodology

Overview

The proposed music variation system is also similar to the evolutionary process. The modification with higher fitness score has higher potential to survive during the selection. Pitch[1], Interval[2] and Duration[3] are the modifiable features and the basic musical components when composing melody. These features are properly represented in MIDI format which is adopted in this study. Different operators are designed to manipulate them and generate variations while the fitness function based on harmony, similarity and diversity directs the evolutionary process. Evolution is an iterative process, after initialization mutations will be performed on the surviving population until satisfying certain fitness criteria.

Genetic Operators

Initialization Genetic algorithm begins its process with the creation of an initial population. Under this scenario, new note sequences that can be composition variations, need to be created. In music variation, a music section is given hence initial pitch and rhythm are already defined. A new note should not be randomly initialized from a huge search space, adding note series or rearranging notes based on the existing music section is more suitable. 1000 new MIDI sequence with modifications on the given music section are created and those with the highest fitness score are selected. Below we describe and illustrate three novel approaches to initialize these new note sequences.

- Split note: This modification splits one note into two of the same pitch with half its original duration. This results in a change in the rhythm while maintaining the same melody contour within the bar (Figure 1).

Figure 1: Split note example.

- Exchange notes: Once the bar is chosen, two selected notes inside the bar will be swapped. The rearrangement of the notes results in a slight variation of the melody (Figure 2).

[1]The basic component in music and can be regarded as notes which represent the European standard system of 12 equally distributed semitones.

[2]The distance between two consecutive notes. Consonant intervals, which sound pleasant during the hearing and dissonant intervals, which create a feeling of tension when hearing.

[3]The length and the timing which one note should occur and finish.The duration of every note in a melody defines the “rhythm”.

Figure 2: Exchange notes example.

- Add note sequence: In order to create a different rhythm, melody and more significant variations, one note is broken into a four related chord degree note sequence (Figure 3).

Figure 3: Add note sequence example.

Crossover Also called recombination, is a genetic operator to combine the information from two candidates. It is a way to generate new solutions from an existing population. There are multiple ways to implement it such as single point crossover, k-point crossover and uniform crossover. As the given music section is too short, performing k-point crossover will not be as beneficial as single point crossover which will favor keeping the better individual notes from good phrases. This process is illustrated by Figure 4.

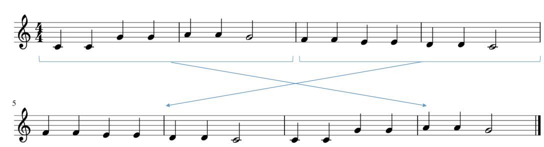

Figure 4: Example of crossover for a given phrase.

Mutation This operator seeks to help the population become better. The mutations occur based on a probability and the positions where they occur are randomly decided. Three mutation alternatives were implemented.

- Pitch Mutation: A note out of harmony will be selected and changed to one of a harmony degree based on the previous note.
- Duration Mutation: A note is randomly selected and its duration is either doubled or reduced to half and. This enables changes in rhythm while keeping the origin melody.

Fitness Rules

Designing the fitness function in GA can be regarded as a critical point since it determines the quality of evolution. Melody similarity, rhythm diversity and harmony are proposed as components of the fitness function. The total fitness score will result from the sum of these three aspects.

Melody Similarity Chord analysis can assist in determining which chords and note series are present in a sequence hence enabling a comparison between two sequences. The Spiral Array Model (Chew and Chen 2005) is adopted to obtain the chord in each bar. Pitches are projected into a 3-dimensional space and every collection of notes is represented by a center of effect(CE), which is a point in the interior of the Spiral Array that is the convex combination of the pitch positions weighted by their respective duration. Consequently, the CE of a bar represents its chord. The score for the distance between the original and generated music pieces can be obtained by equation (1), were b is the total number of bars in the song.

$$S_c = 100 - \sum_{i=0}^{n} (CE_i - CE_i')/b \tag{1}$$

Tempo Diversity To evaluate the differences in tempo we adopt the metrical complexity (Thul and Toussaint 2008). This measurement uses metricity(W), the sum of all the metrical accents of the beats present in a rhythm, to obtain a tempo complexity complexity. Equations 2 and 3 present the tempo diversity score calculation where $maxW_i$ is the maximum metricity for 5 beats and W_i is the actual metricity for every 5 beats in the generated song.

$$TempoComplexity = \sum_{i=1}^{n} maxW_i - \sum_{i=1}^{n} W_i \tag{2}$$

$$S_d = \sum_{i=0}^{n} TempoComplexity_i \tag{3}$$

Harmony For measuring the harmony of a song, the fitness function evaluates every sequence according to music theory. It examines every note from the sequence, whenever a rule is matched, the fitness score is modified accordingly. The higher score means the sequence violates less rules. These rules mainly focus on the basic consonance between consecutive notes and the harmony note with chord sequence, the score arrangement of each rule is listed in Table 1 and its calculation denoted by equation (4).

$$S_h = \sum_{i=0}^{n} (Rule_i * d_i)/n \tag{4}$$

Experiments

Measurement Validation

First, to verify the designed harmony rules can scale the song correctly or not, six testing songs including three from major tonal and three from the minor tonal are evaluated. The harmony score is calculated to see if the proposed rules can scale the harmony degree accordingly. The results presented in Table 2 support theoretical agreements stating harmony is dependent on Major tonality.

No	Rule	Weight
1	2 consecutive notes are the same (C, C)	+3
2	2 consecutive notes are Major 2nd (C, D)	+3
3	2 consecutive notes are Major 3rd (C, E)	+3
4	2 consecutive notes are Perfect fourth (C, F)	+3
5	2 consecutive notes are Perfect fifth (C, G)	+3
6	Big jump between notes (degree $>$ 5)	-8
7	The note is a chord root note	+5
8	The note is a second chord note	+4
9	The note is a third chord note	+4
10	The note is in the scale (C major)	+2

Table 1: Fitness Score for Every Rule

The fitness function was tested on a human composed music variation. "12 variation on Twinkle Twinkle Little Star" consists of improvisations on each section composed by Mozart. Table 3 presents the resulting scores for each varying section. This experiment demonstrates the proposed measurements can properly model the target characteristics pertinent to music variation.

No	Song	Tonal	Harmony Score
1	Through the Arbor	Major	106.03
2	Little Star （Mozart）	Major	99.8
3	Minuet G (Bach)	Major	100.2
4	Sonata No. 5	Minor	66
5	Concerto 5 (Beethoven)	Minor	68
6	Concerto 23 (Mozart)	Minor	71

Table 2: Harmony Score in Different Type of Music

Section	Harmony	Similarity	Diversity
1	100.88	100	0
2	101.33	96.8	5
3	89.8	83.25	5
4	89.33	92.5	8.6
5	87.79	84.88	8.2

Table 3: Harmony and Similarity Score for Little Star Composed by Mozart.

Music Variation Results

The proposed system was tested with songs from multiple genres ranging from classical music to pop. Parameters for the GA were set at 8 bar sequence length, population size of 100, 0.5 crossover and 0.05 mutation running for 600 generations. A demo of the results is available at "https://sharon1018.github.io/". To evaluate the impact of each of the proposed measurements on the fitness function three different variations are presented for each song, incrementally adding an additional component. Samples of the music scores for original an variation songs are presented in Figures 5 and 6.

Through visual inspection and listening to the demo it can be perceived how the musicality is preserved. It is also a

Figure 5: Original melody for Little Star.

Figure 6: Variation Result for Little Star.

good reference to the impact of the different proposed operators and their impact on the evolution.

Human Evaluation of the Variation Results

To rate the quality of the generated songs human evaluation was performed. A total of 20 test subjects, university students aged 22 to 26 with no specific music background, scored the variation results for the different combinations of generated songs as presented in the demo. Figure 7 shows the average harmony score rating, all samples score above 6/10 indicating the songs preserve and even improve their harmonious musical quality while integrating the new features. The samples were also rated in terms of similarity and diversity displaying similar results.

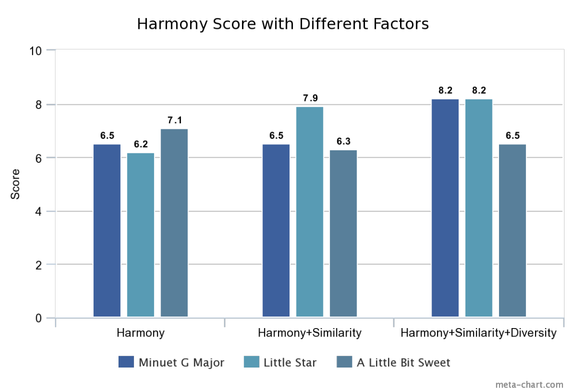

Figure 7: Harmony score for different measurements.

Conclusion

The automated evolutionary approach for music variation is discussed and evaluated in this research. The proposed operators and fitness function guide the evolution process and are able to generate coherent music pieces. The system successfully generated samples that achieve the desired surprise factor in variations and still being relatable to the original songs.

References

Alfonseca, M.; Cebrián Ramos, M.; and Ortega, A. 2006. Testing genetic algorithm, recombination strategies and the normalized compression distance for computer-generated music.

Biles, J. A., et al. Genjam: A genetic algorithm for generating jazz solos.

Brunner, G.; Wang, Y.; Wattenhofer, R.; and Wiesendanger, J. 2017. Jambot: Music theory aware chord based generation of polyphonic music with lstms. In *Tools with Artificial Intelligence (ICTAI), 2017 IEEE 29th International Conference on*, 519–526. IEEE.

Chew, E., and Chen, Y.-C. 2005. Real-time pitch spelling using the spiral array. *Computer Music Journal* 29(2):61–76.

Chu, H.; Urtasun, R.; and Fidler, S. 2016. Song from pi: A musically plausible network for pop music generation. *arXiv preprint arXiv:1611.03477*.

Liu, C.-H., and Ting, C.-K. 2017. Fusing flamenco and argentine tango by evolutionary composition. In *Evolutionary Computation (CEC), 2017 IEEE Congress on*, 2645–2652. IEEE.

Luo, J.; Yang, X.; Ji, S.; and Li, J. 2020. Mg-vae: Deep chinese folk songs generation with specific regional styles. In *Proceedings of the 7th Conference on Sound and Music Technology (CSMT)*, 93–106. Springer.

Majumder, S., and Smith, B. D. 2018. Music recombination using a genetic algorithm. ICMA.

Özcan, E., and Erçal, T. 2007. A genetic algorithm for generating improvised music. In *International Conference on Artificial Evolution (Evolution Artificielle)*, 266–277. Springer.

Sturm, B. L.; Santos, J. F.; Ben-Tal, O.; and Korshunova, I. 2016. Music transcription modelling and composition using deep learning. *arXiv preprint arXiv:1604.08723*.

Thul, E., and Toussaint, G. T. 2008. On the relation between rhythm complexity measures and human rhythmic performance. In *Proceedings of the 2008 C3S2E conference*, 199–204.

Yang, L.-C.; Chou, S.-Y.; and Yang, Y.-H. 2017. Midinet: A convolutional generative adversarial network for symbolic-domain music generation. *arXiv preprint arXiv:1703.10847*.

Yu, L.; Zhang, W.; Wang, J.; and Yu, Y. 2017. Seqgan: Sequence generative adversarial nets with policy gradient. In *AAAI*, 2852–2858.

Zheng, X.; Li, D.; Wang, L.; Shen, L.; Gao, Y.; and Zhu, Y. 2017. An automatic composition model of chinese folk music. In *AIP Conference Proceedings*, volume 1820, 080003. AIP Publishing.