Computer-assisted Music Composition Algorithm Design Dependent on Interactive Genetic Algorithm with Interval Fitness

Qili Guo*
School of Music, Suihua University, Suihua, 152000, Heilongjiang, China

*Corresponding author e-mail: Qili76@su.edu.cn

Abstract. Computer-assisted music composition refers to computer-assisted music composition with the participation of people. However, there are problems such as style and expression. In this paper, a computer-assisted music composition algorithm based on the interactive genetic algorithm with interval fitness is proposed. A new music prediction model is established by integrating melody units and rhythms into traditional models with only notes or rhythms as units. Moreover, the generated music phrases are optimized by the interactive genetic algorithmphrase. The simulation results suggest that the proposed algorithm can generate music phrases quickly with a certain melody logic that conforms to the personal demand of users using a small data set.

Keywords: Computer-Assisted Music Composition, Interactive Genetic Algorithm with Interval Fitness, Genetic Algorithm, Melody Unit

1. Introduction

Computer-assisted music composition refers to computer-assisted composing music composition with the participation of people. Based on the degree of human participation, it can be divided into unitary intelligence and high-level intelligence [1-2]. Unitary intelligence refers to people directly participating in music creation. Computer-assisted music composition is constantly induced. High-level intelligence refers to the indirect involvement of people in music creation, influencing the final generated music through means such as equationing rules or testing results [3]. Algorithmic Music composition Rules combine multiple pieces of music into a whole organic series of rules. Algorithmic music composition does not necessarily require a computer. In the classical period, Mozart used random methods to combine different music modules to create a "Musical Dice Game", and achieved excellent results [4-5].

Markov is a relatively simple and straightforward technique in computer-assisted music composition algorithms. Markov conversion table music composition mainly focuses on connecting single notes or chords or even melody [6]. In fractal music, fractal geometry is applied in intelligent music composition. Based on the principle of self-similarity, synthetic music is created with self-similar phrasephrases. The theme is repeated in a repeating cycle of three and five times with a minor key, which can be used in terms of rhythm. With some random changes, the effects it creates, but
simultaneously, in practical applications, the effects of fractal music have yet to be studied [7]. Artificial neural networks need to collect a large number of works for training. Hence, artificial neural network technology is more suitable for analyzing music works than creating ones. Meanwhile, compared with other methods, artificial neural networks can only create some simpler melodies [8].

Currently, the main technical bottleneck of computer-assisted music composition is about the problems in music creation style, the expression of music knowledge, creativity, human-computer interaction, and the evaluation of musical works. To address these problems, an interactive genetic algorithm with interval fitness is used in this paper to model the melody and rhythm and adds the concept of melody, learn the same method of genre works, establish human-computer interaction, and optimize the results.

2. Design of Music Creation Model Based on Interactive Genetic Algorithm with Interval Fitness

In this paper, the interactive genetic algorithm with interval fitness is used as related factors to describe the relationship between the rhythm and the melody of music. The general steps of music creation are shown in Figure 1 below.

HMM is a statistical model in the time domain, which is often used to describe a Markov process with hidden, unknown parameters. In a normal Markov process, the state is directly visible to the observer. The transition probability between states is all the parameters of the model. In HMM, the state is not directly visible. Instead, some variables affected by the state are visible. Each state has a probability distribution on the possible output symbols. Hence, the sequence of output symbols can reveal some information about the state sequence. HMM contains two probability matrices, the transition probability matrix in the state and the emission probability matrix corresponding to a symbol generated or received upon the state transition. HMM contains 5 parameters below:

(1) The implicit state is expressed in equation (1):

\[ X = \{x_1, x_2, \cdots, x_n\} \]  

Where n is the number of all possible states;

(2) The set of observation symbols is expressed in equation (2):

\[ Y = \{y_1, y_2, \cdots, y_m\} \]
Where $m$ is the number of possible observation symbols corresponding to each state;

(3) The state transition matrix is expressed in equation (3):

$$A = \{a_{ij}\}$$  \hspace{1cm} (3)

$$a_{ij} = p(q_{t+1} = x_j|q_t = x_i), 1 \leq i, j \leq n$$  \hspace{1cm} (4)

Where $q_t$ is the state at time $t$;

(4) The emission matrix, or the observation probability matrix, is expressed in equation (5):

$$B = \{b_j(k)\}, 1 \leq k \leq m, 1 \leq i \leq n$$  \hspace{1cm} (5)

$$b_j(k) = p(o_i = y_t|q_t = x_j)$$  \hspace{1cm} (6)

Where $o_t$ is the observed value of state $x_i$ at time $t$;

(5) Distributed initial state (7):

$$\pi = \{\pi_i\}, 1 \leq i \leq n$$  \hspace{1cm} (7)

$$\pi_i = p(q_1 = s_i)$$  \hspace{1cm} (8)

A simple HMM state transition is shown in Figure 2 below:

![HMM state transition diagram](image)

**Figure 2.** HMM state transition diagram

Hence, an HMM model can be used to describe the state transition process of an unknown hidden state in a known observation state. When the parameters of the model are known, given the observation sequence $O = o_1, o_2, \ldots, o_t$, a most probable state sequence $Q = q_1, q_2, \ldots, q_t$ is selected, that is, the decoding problem in HMM. In this paper, the Viterbi algorithm is used to solve the issue.

2.1. Design of HMM creation model based on rhythm-melody unit

Rhythm is the backbone of music. Even if a melody fluctuates gracefully, without a proper rhythm as the foundation, the melody will become chaotic. HMM offers a good solution to the problem of connecting rhythm and melody. This paper takes rhythm as the observed state of HMM and melody as the hidden state in HMM. Firstly, a rhythm sequence is created. Based on this observation sequence, the Viterbi algorithm is used to generate a new melody sequence.

In the previous studies of ordinary Markov music composition, a single note or note was used as the state space of a random process in most cases. For example, and the frequency of the state of each note is calculated at the next moment as the current state to jump to the next state. This method can only reflect the surface structure of the learning samples. It is necessary to establish a higher-order Markov chain to express the deep structure, which will increase the complexity of the algorithm and
often outweigh the gain. To address this issue, this paper adds the concept of melody yuan to enhance the capacity of HMM to express musical characteristics.

2.2. Training of HMM parameters

In summary, we have already determined that the hidden state of HMM includes a single note in the music to be learned and the extracted melody unit. Based on the definition of this paper, the rhythm of the music to be learned is an observer state. Three parameters to be determined include the state transition probability matrix, the emission probability matrix, and the initial probability matrix.

After the hidden state of the model is determined, the number of times that all states appear in the music phrase to be learned can be counted. All possible states are counted after a state (note or melody unit) and the frequency of these states as the transition probability matrix between states. As equation (9):

\[ a_{ij} = \frac{N(q_j|q_i)}{\sum_{k=1}^{n} N(q_k|q_i)} \]

(9)

Where \( 1 \leq j \leq n \), \( n \) represents the number of all possible next states of the current state \( q_i \). \( N(q_k|q_i) \) represents the number of times that the next state \( q_k \) of the current state \( q_i \) appears. For some states, the transition probability represents regarded as zero if another state does not appear. For example, in “ABCABCDE”, the state “A” does not jump to the state “D”. Then the transition probability from state “A” to state “D” represents recorded as zero.

The emission probability matrix is the probability from a specific hidden state to an observed state. We define the note as the hidden state, and the time corresponding to the note is the observation state. Thus, the emission probability matrix is used to count all possible time values of a note in the music to be learned and the frequency of their appearance. The calculation equation is the same as the state transition probability matrix, as shown in equation (10):

\[ b_{ij} = \frac{N(o_j|q_i)}{\sum_{k=1}^{n} N(o_k|q_i)} \]

(10)

The initial state distribution determines the initial state of the model. In this paper, the first appearance position and the number of appearances of all notes in each music phrase to be learned are counted separately. Subsequently, the initial probability of a specific note is equation (11):

\[ \pi_i = \frac{N(q_i) + \frac{1}{\text{index}(q_{i,i})}}{\sum_{j=1}^{n} N(q_j) + \sum_{j=1}^{n} \frac{1}{\text{index}(q_{j,i})}} \]

(11)

Where \( 1 \leq i \leq n \), \( n \) represents the size of the model state space, and \( N(q_i) \) represents the number of times the state \( q_i \) appears in the learning sample. Imolex(qi) represents the position where the state first appears in the learning sample set. The initial probability of a state represents not only proportional to its frequency in the sample set but also proportional to its position in a sample, which can better reflect the law of performance of the states in the samples set.

Due to the remarkable surface structure of the rhythm of music, it usually presents periodicity and stability. Hence, the first-order Markov chain of the observed state of the model is initialized in this paper. With a single duration in the sample set as the state space and the appearance frequency of the
state as the initial distribution probability, the state transition matrix is calculated. To simplify the algorithm, the length of the state sequence generated by the model is limited to 50.

The HMM-based music composition algorithm proposed in this paper can theoretically generate music works with similar styles to the learning samples. However, music is a product with a strong subjective feeling. Various people may have different feelings about songs of the same genre. Hence, how to enable the music composition algorithm to generate pieces that meet user requirements is also an issue to be considered.

To address this problem, the interactive genetic algorithm is used in this paper. On the one hand, the crossover operator of the genetic algorithm can retain the excellent pieces in the generated music phrases. The mutation operator can make the sudden music change, just like the composer's creative behavior in composing, both drawing on excellent works and exerting inspiration. On the other hand, the interactive genetic algorithm adopts the artificial evaluation function so that the evolution of the music can follow the intention of users to achieve the goal of meeting user requirements. The music coding adopts MIDI real number coding. Given the learning samples selected in this paper, the rhythm coding is limited to one-36th note to full-note. The whole note duration is defined as unit one, and the rhythm code set is: [1/32,1/16,1/8,1/4,1/2,1]. Finally, a piece of music is encoded into a matrix.

The operator and crossover operator are selected in the roulette selection method and multipoint crossover, respectively. In the mutation operator, the following three mutation methods are used.

1) A phrase of the pitch in a chromosome increases or decreases by n steps simultaneously, and the range of n change is set to [1,2,3].

2) A period of rhythm in a specific chromosome increases or decreases by k beats simultaneously, and the value range of k is [0.25,1,1.5,2].

3) Single point random mutation

Upon sudden change, the note or rhythm value out of bounds is randomly initialized to a certain state in the respective state space.

In this paper, manual evaluation is used in the fitness function to assign the fitness value of the music and lead the direction of algorithm optimization. To reduce the audition workload of users in the process of algorithm evolution, the individual fitness value is defined as the weighted sum of the artificial evaluation value and the similarity based on the morphological characteristics of the music.

Where n represents evolutionary algebra, k $\in \mathbb{N}$, $\mathbb{N}$ represents a natural number, $\lambda$ and $\beta$ are positive integers, 20 and 4 are respectively taken in this paper, which can ensure that manual evaluation can completely dominate the direction of evolution along with evolutionary algebra.

3. Experimental Design and Result Analysis

In this paper, a simulation experiment is designed based on MATLAB. The training samples include “Blue and White Porcelain”, “Canghai Qingzhou”, “Legend” and “One Thought for Life”. The algorithm parameters are designed as follows. The length of the generated work is 50 notes. The length of the melody unit extracted based on the dictionary tree search is [2,5], with IGA population size of 10, maximum evolutionary generations of 50, crossover probability of 0.8, and mutation probability of 0.2. In equation (19), $\lambda$ is 20 and $\beta$ is 4, and the range of n in equation (16) is set to [1,2,3], the value range of k in equation (17) is [0.25,1,1.5,2]. Run ten times, and the optimal phrase is shown in Figure 3 below:

![Figure 3. Optimal works](image)

The style of the clip is similar to the learning sample, the rhythm is gentle, and it has a pronounced sense of melody, which is pleasant to the ears and meets the needs of users. In addition, the average population fitness evaluated by users in 20 runs is shown in Table 1 below:
Table 1. Fitness function values of the 20 simulation manual evaluation

|    | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   | 9   | 10  |
|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| H(1) | 4.50 | 3.07 | 6.01 | 2.02 | 7.00 | 3.44 | 5.45 | 3.93 | 4.81 | 3.83 |
| H(2) | 5.21 | 4.49 | 7.03 | 3.61 | 7.91 | 4.81 | 7.32 | 8.01 | 4.92 | 4.31 |
| H(3) | 8.02 | 6.38 | 7.51 | 4.71 | 9.03 | 6.62 | 8.51 | 8.22 | 5.81 | 6.52 |
| H(4) | 8.51 | 8.02 | 8.99 | 5.01 | 9.91 | 8.01 | 9.23 | 9.51 | 7.62 | 8.41 |
| H(5) | 11  | 12  | 13  | 14  | 15  | 16  | 17  | 18  | 19  | 20  |
| H(6) | 3.51 | 2.59 | 4.69 | 5.21 | 4.41 | 3.62 | 4.33 | 5.34 | 4.61 | 5.82 |
| H(7) | 6.11 | 4.32 | 6.71 | 6.83 | 5.84 | 5.81 | 5.62 | 7.83 | 6.82 | 8.01 |
| H(8) | 7.80 | 6.11 | 7.01 | 7.42 | 7.64 | 7.72 | 8.11 | 8.42 | 8.42 | 8.61 |
| H(9) | 8.22 | 7.92 | 8.42 | 8.91 | 8.74 | 9.25 | 9.21 | 8.91 | 9.01 | 9.32 |

As the neutralization in equation (10) is 20 and 4, there should be 4 manual evaluation results in one simulation. Each column in Table 1 indicates the fitness value of 4 manual evaluations in one simulation. Except for the fourth simulation, the algorithm can obtain satisfactory results for users. The works shown in Figure 3 are the optimal results obtained in the fifth simulation.

Compared with Markov’s music composition algorithm, the proposed algorithm has not only taken the possibility of isolation of notes or rhythms into consideration but also established the mapping between notes and rhythms and melody units and rhythms in combination with melody units. The generated works are more coherent and harmonious, with better performance in rhythm. Compared with the other mainstream algorithms based on neural networks, the proposed algorithm has the advantages of high learning speed, small sample size, and effective human-computer interaction.

4. Conclusion

The simulation results have demonstrated that the new computer-assisted music composition algorithm based on the interactive genetic algorithm with interval fitness proposed in this paper can generate phrases of works with a similar style to the training samples by learning them in the early stage, guiding the evolutionary direction of the algorithm based on user demand in the later stage, and finally generating works to the user’s satisfaction. Moreover, the algorithm has low complexity and requires small training samples.

Acknowledgments

This paper is supported by the art and science planning project of Heilongjiang Province, Research on the Nationalization Style of Chinese Pentatonic Mode and Harmonization (Project No.: 2017D010). Project leader: Qili Guo.

References

[1] Feller R. rjan Sandred: The Musical Fundamentals of Computer Assisted Composition rjan Sandred: The Musical Fundamentals of Computer Assisted Composition. By RossFeller. (Gambier, Ohio, USA.). Computer Music Journal, 2018, 42(2):67-71.
[2] Li X, Xie K, Hong W, et al. Hierarchical fragmented image reassembly using a bundle-of-superpixel representation. Computer Aided Geometric Design, 2019, 71(6):220-230.
[3] Bo, Fan, Yunfei, et al. Design of multi-drug combinations for poly-pharmacological effects using composition-activity relationship modeling and multi-objective optimization approach: Application in traditional Chinese medicine.. Chemical Biology & Drug Design, 2018,3(2):1-12.
[4] Popp C J, St-Jules D E, Hu L, et al. The rationale and design of the personal diet study, a randomized clinical trial evaluating a personalized approach to weight loss in individuals with pre-diabetes and early-stage type 2 diabetes. Contemporary Clinical Trials, 2019, 79(2):80-88.
[5] Wang E, Yu Z, Collings P. Dynamic control strategy of a distillation system for a composition-adjustable organic Rankine cycle. Energy, 2018, 141(1):1038-1051.
[6] Xingchen L, Vadim S. Multiscale shape – material modeling by composition. Computer-Aided Design, 2018, 102(3):194-203.

[7] Acuna V V, Hopper R M, Yoder R J. Computer-Aided Drug Design for the Organic Chemistry Laboratory Using Accessible Molecular Modeling Tools. Journal of chemical education, 2020, 97(3):760-763.

[8] Chen Y, Koumaditi E, Gani R, et al. Computer-aided design of ionic liquids for hybrid process schemes. Computers & Chemical Engineering, 2019, 130(2):1-17.