Music Generation using Genetic Algorithm

B.Vijay Kumar[1], Sai Hruday P[2], Farzan Khan[3], Gouthami G[4]
[1,2,3,4] *Member, Dept. of Information Technology JB Institute of Engineering And Technology*

***Abstract*— The gradual implementation of computer music production software within the current stage of music composition and traditional composition methods for integration is essential for the creation of excellent musical works with unique styles and novel ideas. Based on the original composition process, we actively promote the organic integration of music production software and music composition by learning to use computer music production software and drawing on outstanding foreign works. We are constantly discovering inspirations that emerge from music creation, focusing on self-imposed requirements and quality control of music quality, and reducing the cost of music production in the traditional model. With extensive programming technologies, we propose to generate music by coding with a genetic algorithm-based approach which is a search heuristic that is inspired by Charles Darwin's theory of natural evolution. This algorithm reflects the process of natural selection where the fittest individuals are selected for reproduction to produce offspring of the next generation.**

Index Terms: **music composition, fitness, genome, genetic algorithm, MIDI.**

INTRODUCTION

Profoundly, the music market is already suffering from a lack of technical progress. With facial recognition, red-eye reduction, and auto-stabilization procedures, several multimedia areas, such as photography and cinema, have welcomed new technologies. In the same manner, few of the other improvements in music production have become industry-standard tools. Within the sphere of music creation, there is a lot of room for technical progress.

A mode of creative expression humans have embraced in various forms for thousands of years, music has always been reflective of its environment - culturally, politically and, as is increasingly the case these days, technologically

AI backbones a variety of notable music applications today. It generates ever-shifting sound scapes designed for relaxation and focus, powers recommendation systems in streaming services, helps smooth audio mixing and mastering, and generates rights-free music that content creators can use without copyright-clearance hassle. It’s also maturing as a compositional tool, with artists leveraging AI to create new sounds derived from all manner of audio inputs.

Taking a step back from manual production, many artists now employ artificial intelligence in the tune-making process. From composition apps and mastering platforms to song-identifying tools and highly personalised playlists, AI is changing the way music is created and heard. Ease of Use.

The process of turning a musical idea into a piece of music audio that can be distributed and enjoyed is a technical, creative, time-consuming, and cumbersome task. This process referred to as music production, can be broken down into six distinct stages namely song writing, arranging, tracking, editing, mixing, and mastering. Over the last few years, there has been a rise in the availability of affordable equipment, tools and software to carry out various parts of this process. This, coupled with the boom in social media and expansion of music distribution channels has made it very easy for artists and creatives to put their work out into the world. However, irrespective of all the favourable conditions, the hard reality is that the skills and level of expertise required for working with these tools have somehow remained largely the same. These tasks are not only creative but highly technical, hence, requiring years of practice and mastery. This calls for the development of a technology that puts the musician/creative at the forefront and eases the workflow for them. Hence, there is a great requirement for tools that intelligently make the engineering aspects of the content development tasks easier and swift for novice creators, so that they can mainly focus on the creative aspects of the tasks.

It is also understood that music production involves a lot of steps and processes that are quite repetitive and time-consuming in nature. Hence, these tools will also come in handy for a professional who can leave the initial repetitive and technical tasks in the hands of a machine and focus on the more creative and artistic parts of the production process. This could help them explore more ideas in a short period. These tools can also be particularly useful in music restoration and teaching spaces. Finally, there is also a possibility of co-creativity with a machine where the machine can offer ideas and novel perspectives, especially handy when the professional or an artist has hit a creative block.

LITERATURE SURVEY

D. Goldberg [1] in his paper on Genetic Algorithms in Search, Optimization and Machine Learning states, Genetic algorithms are probabilistic search procedures designed to work on large spaces involving states that can be represented by strings. These methods are inherently parallel, using a distributed set of samples from the space (a population of strings) to generate a new set of samples. Although there are several different types of genetics-based machine learning systems, this article [1] concentrates on classifier systems and their derivatives.

N. Tokui and H. Iba, [2] “Music Composition with Interactive Evolutionary Computation,” in Proc. Generative Art2000, the 3rd International Conference on Generative Art [2] Describes a new approach to the music composition, more precisely the composition of rhythms, using IEC. The main feature of our method is to combine genetic algorithms (GA) and genetic programming (GP). In the [2] system, GA individuals represent short pieces of rhythmic patterns, while GP individuals express how these patterns are arranged in terms of their functions. Both populations evolved interactively through the user's evaluation.

Johanson, B. E. [3] 1997 in his paper on the GP-Music System: Interactive Genetic Programming for Music Composition Created automatic fitness raters based on neural networks with shared weights trained with the backpropagation algorithm. They give ratings on a 1-100 scale in a similar fashion to a human using the list interface. In normal backpropagation networks, each connection into a node has its weight which is modified by the backpropagation training. The use of shared weights allows the rating of sequences of variable length, which would be a very hard problem using standard neural network topologies.

Poli, R. and Cagnoni, S. [4] In [4]’s paper on Genetic Programming with User-Driven Selection: Experiments on the Evolution of Algorithms for Image Enhancement [4] present an approach to the interactive development of programs for image enhancement with Genetic Programming (GP) based on pseudocolour transformations. In our approach, the user drives GP by deciding which individual should be the winner in tournament selection. The presence of the user does not only allow running GP without a fitness function but it also transforms GP into a very efficient search procedure capable of producing effective solutions to real-life problems in only hundreds of evaluations.[4] used an approach where 3 instead of asking the user to assign a numerical fitness to all the individuals in a population or to directly select the ones to be used to create the next generation (the strategy used in most of the papers described in the previous section), we ask the user to influence tournament selection by interactively comparing pairs of solutions and determining the winner.

Biles, J. A. [5] Proposed a genetic algorithm-based model of a novice jazz musician learning to improvise. GenJam maintains hierarchically related populations of melodic ideas that are mapped to specific notes through scales suggested by the chord progression being played. As GenJam plays its solos over the accompaniment of a standard rhythm section, a human mentor gives real-time feedback, which is used to derive fitness values for the individual measures and phrases.

Gibson, P. M. and Byrne, J. A. [6] The aim of this [6]’s (Neurogen, Musical Composition Using Genetic Algorithms and Cooperating Neural Networks) paper is to produce a piece of coherent music that resembles that typically found in traditional hymns. A set of neural networks are used to capture the conceptual ideas that build 'good' music and this knowledge is then used to direct a search for the ultimate composition. Genetic algorithms hold many member states representing partial musical fragments. The neural networks cooperate to produce a heuristic value that represents the worth of each of these musical fragments. This value is then used to evolve better compositions based on fragments with high fitness values. The use of Neural Networks as an evaluation function has proved successful in the guidance of genetic algorithms.

Rosa, A.C. [7] in 1999 ("Sample MIDI files") In this paper, the problem of identifying the melodic track of a MIDI file in imbalanced scenarios is addressed. A polyphonic MIDI file is a digital score that consists of a set of tracks where usually only one of them contains the melody and the remaining tracks hold the accompaniment. This leads to a two-class imbalance problem that, unlike in previous work, is managed by over-sampling the melody class (the minority one) or by under-sampling the accompaniment class (the majority one) until both classes are the same size.

FUNCTIONAL REQUIREMENTS

- Selection of Melody Type
- Note Instructions
- Melody Generations
- Fitness of Computer Generations
- Reproductions
- Generation of MIDI files
- Downloading MIDI files

Selection of Melody Type

The user is given a choice to generate either single note melodies or stacks of notes called chords. Note Instructions The user specifies simple mathematical instructions using understandable semantic labelling that act as a deciding factor as to how melodies get generated.

Melody Generations

The system generates a finite number of musical melodies as per the note instructions for further steps as a part of the model.

Fitness of Computer Generations

The users are given a deciding factor by choosing how to fit a musical generation to this personal/business project. An integer score is given to each generation which is analyzed by the model at later steps.

Reproductions

Upon successful scoring, the model analyses the top of the score list regenerates the melodies using simple cut and combination operations and then produce the top two hits of the process to the user.

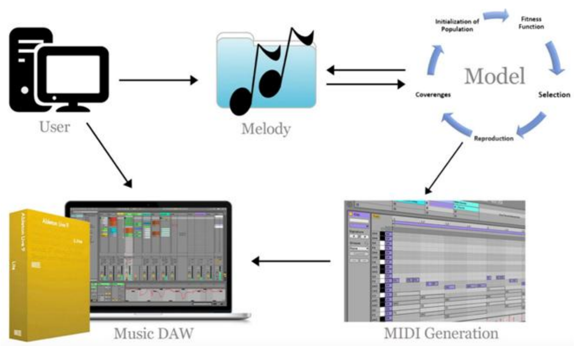
Generating MIDI files

As and when the user decides his final generations, the system then generates the MIDI file and stores it in the local folder.

Downloading MIDI files

Upon the generation of MIDI files, the user then has the access to download those files and use those on different Music DAWs.

SYSTEM DESIGN



ALGORITHM

Genetic Algorithms are a variety of complicated, adaptive algorithm that is commonly used to solve issues involving robust efficiency. In essence, they include working with a population of individuals, each of whom represents a potential solution, and each population represents a subset of the whole search space. In the iterative population process, older individuals are replaced by new, maybe better individuals. Fitness is a value provided to each individual that represents the quality of the thing being observed. Good individuals are picked during the iteration process to (re)produce superior ones using the genetic operator’s crossover and mutation. An older generation is (in some ways) supplanted by a younger one.

Before doing a deep study of the method, the goal and core notion should be stated:

1 The algorithm's goal is to create brief compositions (for example, four 4/4 measures).

2 Compositions are represented by a single array (of integers) that contains pitch and duration information.
3 The general input parameters govern the duration of the composition, tonality, the number and range of tones allowed, the number of iterations, the algorithm's completion criteria, and the manner of interpreting the algorithm's results, among other things.
4 The values that indicate the composition's similarity to the referred composition of the baseline (or reference values), the values of the intervals, and the set of "good" and "bad" tones, allowed deviation (variance) of the prescribed reference values, and weight factors that influence the importance of different assessment criteria are the input parameters that affect the composition quality assessment.
5 Establishing criteria to assess the composition's quality is an important aspect of the algorithm. These criteria concern the length of time between subsequent tones, the divergence from reference values, and the quantity of "poor" tones.
6 The principles of GAs scan the composition search space to identify a composition that is "good enough." It begins with a group of persons chosen at random (compositions). Input parameters govern a portion of the process of producing random compositions. The algorithm aims to locate the individual who fits the criterion to halt the iteration process by applying GA operators from iteration to iteration. The algorithm comes to a halt either when the maximum number of iterations has been reached or when the (best) individual has been generated with sufficient fitness. About the magnitude of the fitness, the individual's quality is inverted. As the individual's fitness (measured in numbers) drops, he or she gets "better."
7 Each iteration calculates the fitness of all individuals in the population, and new individuals are formed through mutations of the currently best ones. Then, among all new individuals and those from the preceding generation, a selection is made.
8 The algorithm's output data is a composition that is regarded as ideal based on the established parameters and iteration procedure.

The Python programming language is used to implement the algorithm.
The initial population, which has n people, is formed according to the starting parameters.
Following that, an iterative procedure begins. Each iteration calculates fitness for each member of the current population. Following that, the population's list of population is ordered by fitness. It is determined if the criterion for the algorithm's finish is reached based on the best one (individual with the greatest fitness). If this is the case, the algorithm will come to a halt, and the best person (composition) will be named as a consequence of the algorithm's execution. If this is not the case, the algorithm begins the process of producing new members. The best individuals are picked from among the present population's members (namely, one-third of the total). After that, mutation operators are applied to them, resulting in the creation of new generations. Each new generation is then added to the existing pool. The updated list of individuals is re-sorted once the mutations are applied to selected individuals (by fitness). The duplicates (generations with the same fitness) are next deleted, followed by the "excess" individuals, to leave exactly n individuals. The iterative process is continued until it meets the termination requirements, such as when the best generation has sufficient fitness or when the algorithm has completed the maximum number of iterations.

CONCLUSION

This study presents a genetic approach to creating musical compositions. The algorithm's output meets several objective requirements for "beautiful" compositions: they contain intervals that are pleasing to the human ear, the rhythm is understandable, and the compositions sound odd but pleasant with a minor change to the proper arrangement. From a practical standpoint, this technique allows you to regulate the numerous aspects that determine the composition's quality and shape. The presence of reference persons (or pre-defined specifications) aids in the selection and creation of a rhythmic and harmonic composition.
Effective and rapid control of the composition, tones, and rhythm is provided by coding the composition with an array of tones and breaks (with extra information about the length). Tones, intervals, and

other "musical" properties can be coded using this coding method, which allows for the application of relevant mathematical functions. It provides numerical values that may execute the arithmetic and logical operations required for any algorithm's execution.

This study may be expanded in several ways. Implementing another metaheuristic for comparison or hybridization with GA would be intriguing. It may be studied how the provided GA could create compositions that all belong to various music genres by altering parameters in the right way.

REFERENCES

[1] D. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning. Reading, MA: Addison-Wesley Professional, 1989.

[2] N. Tokui, H. Iba, “Music Composition with Interactive Evolutionary Computation,” in Proc. Generative Art2000, the 3rd International Conference on Generative Art, Milan, Italy , 2000.

[3] Johanson, B. E. (1997) "The GP-Music System: Interactive Genetic Programming for Music Composition", University of Birmingham, Second-Semester Mini-Project Report.

[4] Poli, R., Cagnoni, S., “Genetic Programming with UserDriven Selection: Experiments on the Evolution of Algorithms for Image Enhancement,” Genetic Programming 1997: Proceedings of the Second Annual Conference, Morgan Kaufmann, 1997.

[5] Biles, J. A. (1994). GenJam: A genetic algorithm for generating jazz solos. In Proceedings of the 1994 International Computer Music Conference, ICMA, San Francisco

[6] Gibson, P. M., Byrne, J. A., “Neurogen, Musical Composition Using Genetic Algorithms and Cooperating Neural Networks,” IEE Conference Publication, No. 349, pp. 309-313, 1991.

[7] Rosa, A.C.(1999)"Sample MIDI files", http:// www.geocities.com/ResearchTriangle/Station/l2 32/ geamusic.html