Towards a Brain Controller Interface for Generating Simple Berlin School Style Music with Interactive Genetic Algorithms

C James-Reynolds[1][0000-0002-5892-5415] and E Currie[2][0000-0003-1186-5547]

[1] Middlesex University, London, UK
[2] Middlesex University, London, UK
C.James-Reynolds@mdx.ac.uk

Abstract. A novel approach to generating music is presented using two interactive Genetic Algorithms with electroencephalogram inputs from two subjects as their fitness functions. Many interactive Genetic Algorithm approaches for generating music employ constrained solution spaces that only utilise notes from a given scale. Our work incorporates the use of mutation to extend the solution space through the inclusion of accidental notes. A thresholding approach is adopted, that allows riffs to be repeated until fitness drops, together with a 'kill switch' to ensure unpleasant sounding riffs are removed from the population.. The development is ongoing, with more testing and calibration required to ensure that there are no timing errors in communication between the microcontroller boards and to identify the most appropriate threshold and mutation ranges, in addition to determining the most appropriate mixes for the users to hear.

Keywords: interactive Genetic Algorithm, Electro-Encephalograph, Music generation.

1 Background

1.1 Literature Review

Direct Brain to Music interfaces have been built [1] but it is not yet possible to map brain output in the form of electroencephalogram (EEG) data for example to create music that can be performed by a synthesiser as it is imagined. A simple interface of this type would map energy in frequency bands from the EEG data to Musical Instrument Digital Interface (MIDI) data for example. Of course it is possible to have an intermediary system that can do this. The human body with its appendages can do this with years of training. Technology allows us to create user interfaces that make it easier to create meaningful music by restricting the user to specific scales, rhythms and patterns; for example the Korg Kaossillator [2] does this. Alternative approaches have made use of evolutionary algorithms. [3,4,5] These are not Direct Brain to Music interfaces, but users can evaluate outputs and adopt the role of a fitness function.

Genetic Algorithms (GAs) are a form of search or hill climbing algorithm based on Darwinian evolution theory. They were popularized by Holland [6]. We typically start off with a random population of solutions where each solution is considered as an individual. The parameters that define the solution are the alleles. We then evaluate the individuals for fitness by using a fitness function. Individuals that have the best fitness are then selected to be parents on the basis of that fitness. Different selection strategies can be used such as ranking and roulette wheel approaches. Parents are then used to create offspring using a crossover function that combines alleles. The process repeats until an optimal solution is generated. To avoid the possibility that the original population did not have sufficient diversity to find an optimal solution, for each of the offspring, random mutations may be applied with a given probability [7].

Interactive GAs differ from tradition GAs in that the fitness function is based on human interaction, typically by a human consciously rating the output. Ratings can also be obtained subconsciously by measuring the subject's response to the output using sensors such as EEG [5, 8,9,10].

iGAs such as GenJam [3] have used crowd sourced input as their fitness function to produce musical scores. This project explores the use of two individual subconscious human inputs using EEG, into two iGAs. One iGA generates lead riffs and the other iGA generates bass riffs. These are outputted together as a single coherent piece of music for rating by the listeners. A threshold function is implemented to keep riffs (We use the definition from [11]) playing by pausing the playing of the next pattern while mindfulness [12] levels remain high and these resume when the mindfulness levels falls below a specific threshold. Typically many iGA strategies allow for a reduced solution space to facilitate the greater probability of producing musically pleasant riffs. This may take the form of using a specific key and scale type to ensure that note sequences are not "unpleasant" sounding. In order to achieve the benefits of this approach and the greater probability of creating genuinely interesting patterns facilitated by a larger solution space, the introduction of accidental notes via mutation is allowed to increase the solution space.

A more restricted solution space could produce solutions similar to those of Mozart's dice game[13,14], which uses the idea of throwing dice to generate a piece of music from pre-selected patterns that fit together. This works with music where some simple structural rules can be applied such as Electronic Dance Music.

1.2 Motivation

The work described in this paper is based on previous work involving the use of an iGA to generate Mondrian-style paintings. [9] and visual effects to assist with achieving mindfulness [10]. The fitness function of these iGAs was computed from real-time EEG readings from the subject, thus quantifying the subject's subconscious responses to the successive 'paintings' and patterns with which they were presented by the iGA. From these works, two issues were identified that needed further exploration. Firstly it was noticed that when EEG readings were used as a fitness function for an iGA, the human subject would often produce a high rating for mindfulness for a while as the algorithm converged on a satisfactory solution (one for which the subject might choose to stop the algorithm), but after a time the rating would drop, possibly indicating that the subject was bored, distracted or that the image no longer held the same appeal.

This is easy to understand in terms of music. A short riff may be catchy and interesting to listen to, but if it is played for a long period of time, it might become boring or even irritating. The introduction of a threshold value for the fitness function that would cause the iGA to pause when it found a good enough solution, store that solution and then to continue when the rating dropped below the threshold. We also considered some of the issues caused by restricting the solution space to a limited set of solutions. For example, in [10] the solution space was limited to values obtained through analysis of a number of Mondrian's works and teasing out a simplified version of their common defining characteristics. The software then embodied these rules, leading to the production of creations with limited variability. Boden discussed this issue in her work on creativity and AI [15].

However, utilising unrestrained solution spaces can also cause interesting issues to arise. For example, in a previous iGA project [16], we found that allowing a wide range of variables in an interactive synthesiser created sounds that were fit solutions inasmuch as they encouraged mindfulness, but were not musically useful. By using a compromise it is hoped to allow the solution space to grow over time if selection permits; this is achieved by enabling mutation to select values (accidental notes) that are outside the initial solution space. This might also produce some unpleasant results and undesired results could be terminated early.

Music is often the result of collaboration and musicians will often "jam" together in order to develop riffs and songs. In order to achieve this, we employ two EEG headsets with two users; one responsible for the lead line and the other responsible for the bass line.

The prototype is being developed on the Arduino platform [17], as this is low cost and standalone.

2 Implementation

Two Mindwave EEG headsets are used [18] to send data to two Bluesmirf Silver Bluetooth [19] modules connected to Arduinos. The headsets provide data as a range from 0 to 100 for different frequency bands, The code parses the data and allows for EEG bands identified by the manufacturer as attention and meditation to be selected via a pushbutton and a corresponding LED displays the selection. The parameters available in the implementation are attention, meditation and an average of the two which is taken to represent mindfulness [20]. In order to facilitate testing and modification of the code, a software based serial link using pins 10 and 11 of the Arduino as Tx and Rx respectively are used to send the data to the Arduinos running the iGA.

The individual solutions generated by the iGA algorithms running on each of the two Arduinos differ in the number of notes in their patterns and in their octaves. This allows one of the iGAs to produce a bassline and the other a lead line. The patterns are also adjusted accordingly.

The iGA has four seeded patterns as the starting parents and these are sent to the MIDI Master Board over I2C. Each parent consists of the following parameters:
- Notes – currently a one octave range (C to C), the bassline is two octaves lower than the lead line, Notes are 7 bit integers.
- Key – for both riffs
- Pattern – Eight patterns with 16 slots are available for the lead line. The bass line has eight patterns with 8 slots allowing timing changes
- Delay Time – delay times are in fractions of the tempo
- Delay Amount – the level of the delay (currently not implemented)

The MIDI Master Board plays a patterns for 4 bars (this is easily adjustable for more or less bars) and if the threshold has been met will continue to play for another 4 bars or until the threshold falls. The tempo of the music is manually controlled by a potentiometer and allows a range of 80 to 140 Beats Per Minute. The MIDI output is sent to two MIDI channels on a synthesiser (Currently using a Roland MU50 to test) and also to an effects unit to create the delay effect. The next sequence is then sent to the MIDI Master Board. At the end of each sequence the current EEG data is stored as a rating value by the IGA Boards. Button switches connected to the iGA Arduinos and the MIDI Master Board allows riffs to be killed off if they are unpleasant, much in the way that musicians jamming might try something and then not repeat it if it does not sound pleasant.

The iGA can use either a roulette or tournament method for selecting candidates for creating the next generation and this can be changed in the code. Offspring are created using a single point crossover and have a chance of mutation for all their values, the mutation probability can be determined in real-time with a potentiometer, currently allowing between 0 and 10%.

One of the challenges in designing the hardware is the communication between different boards. The I2C protocol used does allow for boards to switch between Master and Slave, but this was considered to be overly complex, as only one board at a time can be Master and the EEG Boards and iGA Boards run in parallel. Use of I2C between MIDI Master and Slave iGA boards is appropriate as the communication can always be initiated by the Master which will request data when it is ready to play. A simple serial communication between the EEG boards and their associated iGA boards allows for communication between the slaves and is accomplished using a software serial port. The prototype wired for testing the communication is shown in Fig.1.

Overall structure of iGA algorithm (simplified):

Initial seeded population sent from iGA boards to MIDI board

(A)For each individual ():

> MIDI board plays individual for x number of times (currently 4)
>
> Rating is taken from EEG board as an average of meditation and attention and stored
>
> If rating exceeds threshold for satisfactory solution, repeat from (A)

Ratings of all individuals are used to create roulette wheel

Roulette wheel runs to select parent pairs

Crossover and mutation (approx. 5% used currently) applied to create offspring
Repeat from (A).

An example of an output on the 6^{th} generation is C, C (Oct), A, G, $D^{\#}$, E, F, F where the D# is from a mutation and works musically. Fine tuning of mutation rate and thresholds for each user can be modified on the fly by means of potentiometers.

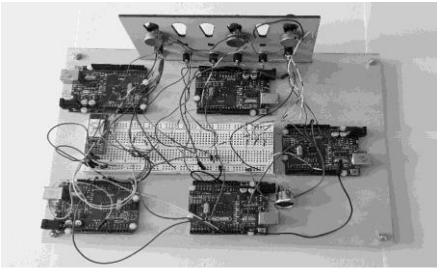

Fig. 1. The prototype for testing communication on the bass line.

3 Limitations and Future Work

The main purpose of the research was to explore the use of thresholding to provide novel and interesting solutions, with the potential of extending the solution space through mutation. The use of the crude EEG headset is a limitation and it might be possible to get much better results with more sophisticated EEG equipment.

From a musical perspective electronic music in this style often evolves quite subtly over time and may revisit earlier riffs, This can be achieved by making the riffs chosen for initial seeding more similar to each other, which leads to the generation of more subtle sequences of changes by the algorithm. It might be possible revisit riffs by allowing "good" riffs to be saved to memory and reintroduced to the population.

It is difficult to be certain whether the current approach is optimal. It might be better to have a single iGA for both the lead and bass riffs and this could also be extended to include rhythm [21]. However, the advantage of having separate users responsible for different instruments is that they can make their own decisions about when to remove an individual from the population or keep it for future use.

Currently the main developmental issues are around the timing and communication between boards. Having delays between notes is not a problem if all the delays are equal. When one note is slightly out of time it disrupts the flow of the sequence.

References

1. Miranda, E. R., & Brouse, A. (2005). Interfacing the Brain Directly with Musical Systems: On Developing Systems for Making Music with Brain Signals. Leonardo 38(4), 331-336.
2. https://www.korg.com/uk/products/dj/kaossilator2s/ Accessed 7 July 2021.
3. Biles, J. A. "GenJam: A Genetic Algorithm for Generation Jazz Solos". Proceedings of Computer Music Conference (ICMC '94), pp. 131- 137, 1994.

4. Miranda, E. R. (2006) 'Brain-computer music interface for generative music', Virtual Reality, p. 8.
5. 'Vox Populi: Evolutionary Computation for Music Evolution.pdf' (no date).
6. Holland, J. H.: Adaptation in Natural and Artificial Systems. Ann Arbor: University of Michigan Press. (1975)
7. Melanie, M. (1999) 'An introduction to genetic algorithms', Cambridge, Massachusetts London, England, Fifth printing.
8. James-Reynolds, C., Currie, E., 2016. EEuGene: Employing Electroencephalograph Signals in the Rating Strategy of a Hardware-Based Interactive Genetic Algorithm, in: Bramer, M., Petridis, M. (Eds.), Research and Development in Intelligent Systems XXXIII. Springer International Publishing, Cham, pp. 343–353. https://doi.org/10.1007/978-3-319-47175-4_25
9. Němečková I., James-Reynolds C., Currie E. (2019) Evolutionary Art with an EEG Fitness Function. In: Bramer M., Petridis M. (eds) Artificial Intelligence XXXVI. SGAI 2019. Lecture Notes in Computer Science, vol 11927. Springer, Cham. https://doi.org/10.1007/978-3-030-348
10. James-Reynolds C., Currie E. (2019) Mindfulness Mirror. In: Bramer M., Petridis M. (eds) Artificial Intelligence XXXVI. SGAI 2019. Lecture Notes in Computer Science, vol 11927. Springer, Cham. https://doi.org/10.1007/978-3-030-34885-4_36
11. Salomoni M., Carter J. (2019) GenMuse: An Evolutionary Creativity Enhancement Tool. In: Bramer M., Petridis M. (eds) Artificial Intelligence XXXVI. SGAI 2019. Lecture Notes in Computer Science, vol 11927. Springer, Cham. https://doi.org/10.1007/978-3-030-34885-4_18
12. Bishop, S.R., n.d. Mindfulness: A Proposed Operational Definition 12. Clinical Psychology: Science and Practice; Autumn 2004; 11, 3; Health Module pg. 230
13. S. A. Hedges. Dice music in the eighteenth century. Music & Letters, 59(2):180–187, 1978.
14. Noguchi, H. 1996. Mozart: Musical game in C K.516f. Available at http://www.asahi-net.or.jp/~rb5hngc/ e/k516f.htm.
15. Boden, M. A. (1998). Creativity and artificial intelligence. Artificial Intelligence, 103, 347–356.
16. James-Reynolds, Carl and Currie, Ed (2015) Eugene: a generic interactive genetic algorithm controller. In: AI-2015: The Thirty-fifth SGAI International Conference, 15th-17th December 2015, Cambridge.
17. Arduino. https://www.arduino.cc/. Accessed 7 July 2021
18. NeuroSky (2018). *NeuroSky Brainwave Starter Kit*. http://developer. neurosky.com /docs/doku.php?id=mindwave_mobile_and_arduino [Online; accessed 19-July- 2021].
19. Bluesmirf https://www.sparkfun.com/products/retired/12577 (last accessed 16/07/21)
20. Lomas, T., Ivtzan, I., Fu, C.H.Y., 2015. A systematic review of the neurophysiology of mindfulness on EEG oscillations. Neuroscience & Biobehavioral Reviews 57, 401–410. https://doi.org/10.1016/j.neubiorev.2015.09.018
21. Horowitz, M. D. (1994). Generating rhythms with genetic algorithms. In Proceedings of the AAAI National Conference on Artificial intelligence, Menlo Park.