Mapping brain signals to music via executable graphs

Katie Crowley
Trinity College Dublin
k.crowley@tcd.ie

James McDermott
University College Dublin
jmmcd@jmmcd.net

ABSTRACT

A method for generating music via a mapping from brain signals is proposed. The brain signals are recorded using consumer-level brain-computer interface equipment. Each time-step in the signal is passed through a directed acyclic graph whose nodes execute simple numerical manipulations. Certain nodes also output MIDI commands, leading to patterned MIDI output. Some interesting music is obtained, and desirable system properties are demonstrated: the music is responsive to changes in input, and a single input signal passed through different graphs leads to similarly-structured outputs.

1. INTRODUCTION

Mappings between sensory modalities are fascinating. Synaesthesia is an example: some people report experiencing certain colours when they hear certain pitches, for example; others report an association between colours and letters [1]. One of the authors recently heard a two-year-old child refer to some intense crayon work as "doing loud on the paper". Feeling the kick drum in your chest is indispensable to some forms of music. Jean-Michel Jarre and many others have made mappings between light and music. Douglas Hofstadter poses questions like, what would a poem be like if it were in the medium of painting instead [2] – and recreational drug users sometimes report answers. In our favourite songs, we often feel that there is an essential link between the words and the music – not just that they are well-suited, but that they are synchronised, with a clear mapping between them at each point in time. Something similar happens with film scores.

The voltages that are produced by the human brain as a by-product of its normal activity are not a sensory modality similar to, say, sight or hearing. Nor are they a modality we have obvious control over, like speech. However, a mapping between the brain's activity and the resulting voltage signals can be established [3]. It is therefore of interest to think about mappings from these signals to other modalities. Because these signals are time series, it is particularly natural to consider mappings to a time-based medium like music. The fact that a feedback loop is possible – brain to signal to music to ear to brain – greatly increases the possibilities and the interest.

In the long term, we hope to use mappings and feedback between brain signals and music for forms of music therapy [4, 5]. In this initial study the goal is much less ambitious: it is to map brain signals to engaging, listenable music which is synchronised with the brain signal and reflects changes in it. It is thus a form of sonification. Success in these initial steps is required for the longer-term goal.

The output of such a mapping will depend on both the input and the mapping itself. However, the goal is to achieve a sort of separation of control between the two. The mapping should be capable of achieving a somewhat listenable, if dull "steady state" of music in response to a completely static input, but should also be responsive to changes in the input. Similarly, a single signal mapped through different mappings should give results which, if not really similar in style or content, are similar in temporal structure.

2. BACKGROUND & PREVIOUS WORK

2.1 Brain-Computer Interfaces

The human brain is made up of billions of neurons, which emit electrical impulses and changes in hemodynamics when interacting. The electrical impulses form a measurable voltage on the scalp that can be detected by electroencephalogram (EEG) devices. A Brain-Computer Interface (BCI) is a system which measures changes in this voltage in real-time [6,7]. Typically the raw EEG signals are pre-processed to produce a usable time-series or in some cases a command output. BCIs have applications in human computer interaction.

Modern BCIs can be non-invasive, portable, low-cost, and easy to use, with high temporal resolution. The cheapest ones may use just one or two EEG sensors fitted to a light-weight headset, in contrast to medical grade BCIs.

2.2 BCI Music

Research into BCIs for music is a growing area with potential in artistic, scientific, recreational and therapeutic fields. The earliest reported example of EEG-based musical analysis was in *Brain* in 1934 [8, cited by Miranda [9]], however, it is generally accepted that EEG-based composition began with Lucier's *Music for Solo Performer*, a percussive piece composed by the performer wearing an EEG cap. Teitelbaum used various physiological signals including EEG and electrocardiogram (ECG) to control electronic synthesisers [10]. Rosenboom also examined the use of EEG signals to generate art, including music, and developed EEG-based musical interfaces [11]. Rosenboom introduced a musical system whose parameters were driven by EEG signals associated with changes in the performer's selective attention [12]. Interested readers may refer to Williams [13] for a comprehensive review of the history of BCI music.

Affective Algorithmic Composition (AAC) is a proposed umbrella term [13] referring to an interdisciplinary field which combines computer-aided composition with affect analysis (or emotion assessment). AAC algorithms are driven by an intended affective response from the listener, who in turn, can become the composer. AAC includes any system for composition designed to respond to an affective target and/or to create an affective response in the listener [13]. A listener might use a bio-signal device to measure some physiological response, for example, to generate affectively responsive music.

Existing work uses brain-computer control systems to allow users to control musical parameters via EEG [14–16]. Miranda et al. [4] describe the evaluation of a pilot brain-computer musical interface allowing a patient with Locked-in syndrome to control amplitude and other musical parameters via EEG for the purposes of music therapy and palliative care [4]. Advances in modern BCIs and non-clinical EEG provide an opportunity to develop more commercially-accessible neurofeedback-derived control over musical features in response to individual affective responses resulting in real-time biophysical sensing of emotions to control AAC systems.

2.3 Other Mappings

Moving away from BCI, one important stream of research in mapping signals to music has arisen in the context of evolutionary computation (EC). EC is a class of population-based metaheuristic search and optimisation algorithms inspired by Darwinian evolution. EC approaches to music usually take advantage of some form of mapping rather than trying to create music directly. An interesting mapping was proposed by Hoover [17]. Time is divided into time-steps. At each time-step, some variables are fed into a neural network. They represent the events in the corresponding time-step of some pre-existing music. The neural network maps these values to produce multiple outputs, which can be interpreted as MIDI commands. By running the network once per time-step, with the input signals varying over time, the result is a new piece of music synchronised with – because it is created as a mapping from – the input piece. Naturally, the mapping must be of sufficient complexity that the output is not a simple monotonic transformation of the input.

This method was used to interactively create drum tracks to accompany pre-existing harmonic and melodic material [17]. The network was trained through interactive evolution, that is via preferences for one network's results over another's, expressed interactively by a listener. An appealing feature of the representation is that in a neural network, computation is "shared" – the same result calculated at one node can be re-used by multiple nodes at the next layer, and so the multiple outputs can be expected to be related. Of course, that is a desirable property for the multiple voices of many types of music.

The same idea was later extended to create pitched accompaniment material, and to also use simple signals indicating the "semantics" of the current time-step – whether it is the start of a beat, and whether it is the start of a bar [18]. "Complex conductors", i.e. arbitrary time series as further input variables, were also proposed.

Inspired by these mappings, the second author has developed [19] a directed acyclic "executable graph" representation which again uses input variables representing the time-step's "semantics", shared computation in the graph, and multiple outputs mapping to MIDI commands. The main differences from NEAT Drummer and subsequent work are: (1) the model of computation does not use an implicit weighted sum of inbound edges at each node. The arity of the function executed by a node determines the number of inbound edges that it requires. (2) No input music is used. (3) The output nodes are *stateful*, that is their inputs and outputs in previous time-steps can affect their outputs in the current time-step. In another implementation, trees (rather than graphs) are used, and input signals are supplied by the user via a mouse or Nintendo Wiimote [20].

These representations are capable of generating listenable music, at least over short time-scales. It is natural to consider using them as general methods for sonifying any type of time series. That is the point we take up in this paper. We propose a mapping and investigate how well it achieves our goals:

- A static BCI input signal should lead to listenable (if dull) steady-state music;
- The music should respond to changes in the input signal;
- The temporal structure of the input signal should be reflected in the output;
- As a consequence, a signal mapped through different graphs should lead to pieces of music which share temporal structure.

We begin by describing more details of the mapping, including novel features not used in previous work.

3. MUSIC WITH EXECUTABLE GRAPHS

3.1 Music as a function of time

The representation is a development of that in *XG* [19], as inspired by that of *NEAT Drummer* [17]. Time is divided into even time-steps, e.g. six steps per quarter-note. At each time-step, the values of some numerical variables (described later) are fed into a graph. The nodes of the graph may output MIDI note-on or note-off messages. In this way, temporal patterns in the input variables give rise, via a mapping, to temporal patterns in the output. In this representation we can think of music as a function of time, and of the input time-series.

3.2 Model of computation

The graph is directed and acyclic. Input nodes carry the input BCI signals. Other nodes have incoming edges and carry out numerical computations such as +, *, or *sin*. The graph is constrained to have the right number of inbound edges to each function. On the other hand, each node may

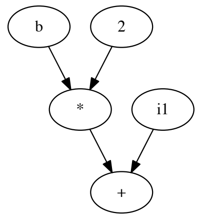
Table 1: Labels, computations, and arities for all node types.

label	result	arity
i_1	input signal 1 at current time-step	0
i_2	input signal 2 at current time-step	0
b	beat at current time-step (an integer)	0
0.5	constant 0.5	0
1	constant 1	0
2	constant 2	0
unary-	$-x$	1
*	$x * y$	2
+	$x + y$	2
-	$x - y$	2
pdiv	$x/\sqrt{1+y^2}$	2
pmod	$x \;\%\; \sqrt{1+y^2}$	2
sin	$\sin(x)$	1
cos	$\cos(x)$	1
if	if $x \geq 0.0$ then y else z	3

send its output to any number of other nodes. In contrast to a neural network, there is no implicit weighted sum of inbound edges' signals.

Because the graph is constrained to be acyclic, the nodes can be sorted using "topological sort", i.e. nodes which have no inputs are placed first, and every node is placed after the nodes which give its inputs. Thus each node can be executed in this order, to execute the entire graph. The graph is thus *executable*. The labels, computations, and arities for all node types are shown in Table 1.

As an example, Fig. 1 shows a very simple graph and its effect. Just one input signal i_1 is used, together with the beat signal b. The values of these signals over six successive time-steps (two bars in 3/4 time) are shown, together with the output of the $+$ node. We can interpret these outputs as pitch values for MIDI note-on commands. (This is a simplified example, with a very small graph, just one input signal, and ignoring the effects of restrictions on output nodes, accumulators and thresholds, and sigmoid and diatonic mappings, to be described in detail below.)



time-step	0	1	2	3	4	5
b	0	1	2	0	1	2
i_1	50	50	50	58	58	58
output (at $+$)	50	52	54	58	60	62
pitch	D	E	F#	A#	C	D'

Figure 1: A simplified example of the mapping process.

3.3 Graph generation

The graph generator starts by creating one node each with the labels `i1`, `i2`, `b`, `0.5`, `1`, and `2`, i.e. all those with arity 0. It then adds 100 nodes, each with a label randomly-chosen from those with non-zero arity. For each node, it adds the appropriate number of inputs (according to arity), taken from the output of any previously-added node. In this way, the property of acyclicity is also guaranteed.

3.4 Output nodes: accumulators and thresholds

The graph as described so far deals with purely numerical values. In order to produce music, these numerical values must be mapped to MIDI note-on/note-off commands. This takes place at output nodes. An output node is just a normal node of the graph, with a label chosen from Table 1 as usual. However only a node with at least two inputs, and such that there is at least one path from an input node to this node of length 3 or greater, will be used as an output node. This multiple-output representation is reminiscent of that used in *single-node genetic programming* [21].

An output node's inputs are used for a numerical computation, determined by its label, and resulting in a numerical output as usual. In addition the inputs are interpreted as pitch and activity controls. An output node has an accumulator variable which is increased by the value of the activity control, via a sigmoid mapping, at each time-step. Whenever the activity is above a numerical threshold (set to 1.25 in experiments reported here), two things happen: a MIDI note-on command is output, with pitch controlled by the pitch control input, via a sigmoid mapping and a diatonic mapping, and with velocity controlled by the degree to which activity exceeds the threshold; and the accumulator variable is decreased to account for this command. Whenever the activity is below a second, lower threshold (0.0625), a MIDI note-off command is issued, for the pitch most recently switched on. Thus pitch, velocity, and note duration are explicitly controlled. Whenever the activity is between these two thresholds, there is no MIDI output, but the activity variable is decreased.

The sigmoid mapping is standard, $x \rightarrow 1/(1 + e^{-x})$. The motivation for the mapping in both pitch and activity is that the output of a node can vary widely, especially with multiplication and division. The mapping "squashes" large values (positive or negative).

These computations lead to quite human-sounding variations in note volume and note density. Although it is still fully deterministic, it tends to avoid the metronomic or robotic feeling that can easily arise in generated music, e.g. to some extent in the output of previous work [20]. The individual voices in the music play and rest and form phrases with pleasant dynamics.

The restriction on input path length for output nodes helps to prevent very simple (e.g. monotonic) transformations of the input from occurring in the output.

There are several important parameters in the representation, including the accumulator threshold and the number of nodes in the graph. The values given above for these parameters have been found to give good results, but investigation of optimal values is postponed to future work.

4. BCI HARDWARE AND PROCESSING

An EEG signal is a voltage that is measured on the surface of the scalp, arising from neural activity e.g. mental state, cognitive activity etc. Fluctuations in the EEG signal occur within defined frequency bands that have been associated with brain states such as attention (Beta: 13–30Hz), engagement, frustration, meditation (Alpha: 8–13Hz) and so on. Changes in the signal within these frequencies bands can be measured by EEG devices, which reflect changes in neural activity. Some of these bands relate to emotion-based responses, and concentrating on these frequencies, we can capture emotional response data.

4.1 NeuroSky MindWave

NeuroSky Technologies have developed a minimally invasive, dry biosensor to read neural activity representing states of attention (Beta) and meditation/relaxation (Alpha). The *MindWave* headset consists of a single dry sensor positioned at the forehead on a position known as FP1, to capture activity from the pre-frontal cortex in the front of the brain where higher thinking occurs. Emotions, mental states, concentration, etc. are all dominant in this area. The MindWave captures raw neural signals at FP1 and provides information on a user's Alpha, Beta, Gamma, Delta and Theta bands. The signals are captured at 512Hz, filtered and processed using a Fourier transform, and passed to a proprietary algorithm which generates *eSense* values, custom measures of attention and meditation [22]. For each of attention and meditation, the algorithm returns one value per second on a scale from 0 to 100, representing the level of attention or meditation of the subject [23].

In a previous study, Crowley et al. [24] identified threshold values for these eSense scales in order to categorise response intensity. Using these threshold measures, an eSense value between 40 to 60 at any given moment in time is considered "neutral". A value from 60 to 80 is considered "slightly elevated", and values from 80 to 100 are considered "elevated". Similarly, on the other end of the scale, a value between 20 to 40 indicates "reduced" level of response, while a value between 1 to 20 indicates "strongly lowered" levels.

The meditation value returned by the headset is used to record the users' state of arousal, which indicates the level of a user's mental "calmness" or "relaxation". If the user is relaxed and not under stress then the value returned is high (high meditation = low stress). The eSense Attention meter indicates the intensity of a user's level of mental "focus" or "attention", such as that which occurs during intense concentration and directed (but stable) mental activity. The attention value captures the users' level of effort. If the user's effort level is high then the output can near 100 whereas if they make no effort at all it is nearer 0 [24]. While the headset records both the raw EEG signal and the eSense measures, our analysis focuses on the custom attention and meditation scales for their potential as easy-to-use, "off-the-shelf" measures of EEG signal activity that could be used by signal processing novices.

4.2 BCI Data Collection and Preparation

To produce the input BCI data for the system a number of tasks were completed. The aim was to use both baseline and task-related BCI data as inputs for the system. A subject was fitted with the NeuroSky MindWave device and asked to complete a number of tasks while wearing the BCI headset. Firstly, the subject was asked to sit quietly for 5 minutes while baseline recordings were measured. Three *stressor* tasks were then administered – The Towers of Hanoi, an N-Back Task and an electric wire loop game. These are common in psychological and BCI research as described, e.g. by Crowley [24, 25]. These are not musical tasks, hence the system is functioning as a sonification of the BCI data rather than a method for the subject to control music.

The three stressor tasks produced BCI data that varied in attention and meditation levels from baseline. Each task is designed to elicit varying degrees of stress (low meditation) and require different amounts of cognitive load (attention) depending on the individual response of the participant. The eSense meters of *attention* and *meditation* for each task were extracted from the BCI recordings and used as inputs i_1 and i_2 to the executable graph.

Several composite signals were created, with the goal of imposing clear temporal structure:

ABA means A signal of 48s in ABA format, where A uses the mean value of the subject's baseline recording for 16s, and B uses the mean value of the subject's Towers of Hanoi recording (lowered meditation and raised attention) for 16s.

ABA non-means A similar signal in ABA format, but using 16s of raw signal from the baseline and Towers of Hanoi recordings, rather than means.

ABACADA non-means A similar signal in ABACADA format, where C and D are raw signal from the N-Back and Wire Loop tasks (both tasks again leading to lowered meditation and raised attention).

5. RESULTS

The executable graph mapping was used to generate many pieces using various BCI signals. Here we concentrate on pieces made using two different graphs, and using the composite signals described above. The "ABA means" signals were used to demonstrate that a static input signal leads to a static musical pattern. The simplicity of the output then made it suitable for use during auditioning of multiple graphs. We chose two graphs which led to interesting patterns, corresponding to random seeds numbers 1 and 5. The latter graph is shown in Fig. 2. For the former we chose a minor scale mapping [1] and for the latter, a major scale.

The "ABA non-means" signals were then used to investigate the result of non-static input signals. The results were encouraging: the non-static input signals introduce variation, but not so much that the piece loses a sense of close similarity with the "means" version.

[1] Refer to the subject2_means_aba_seed1 mp3 available for download.

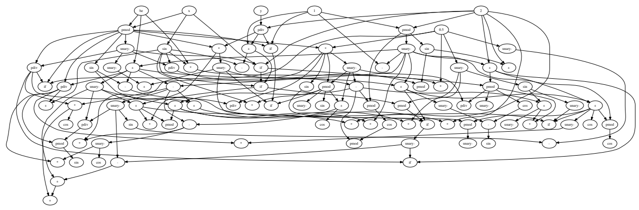


Figure 2: One of our chosen graphs. Due to its size (100 nodes) the labels are not readable here, but the graph is available online.

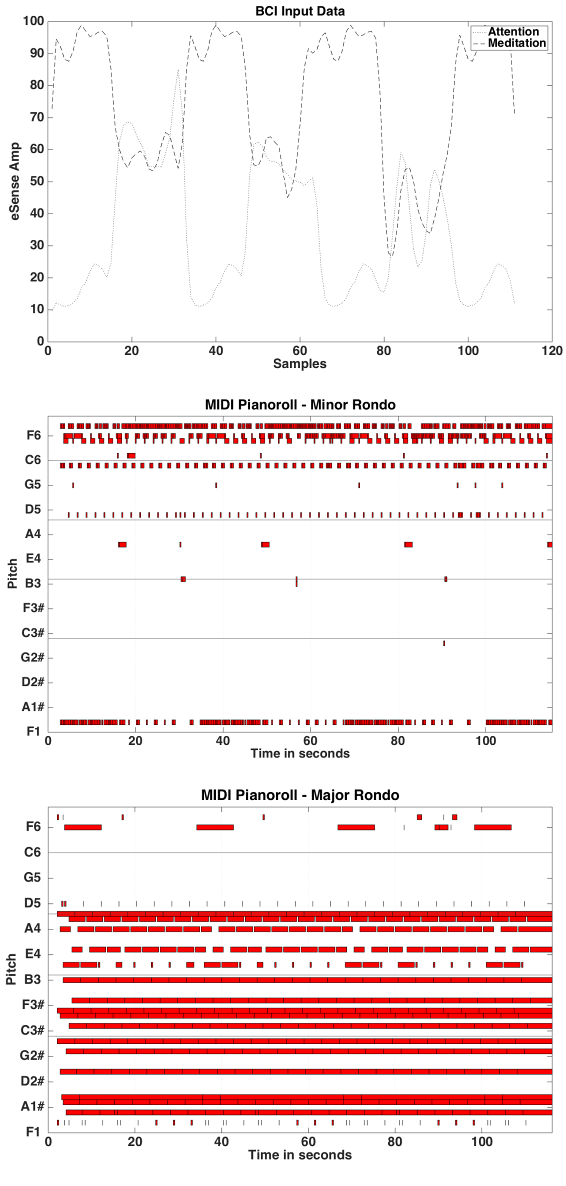


Figure 3: A BCI input (with two signals) in *ABACADA* form and two distinct pieces of music, with the same temporal structure, resulting from mapping this input via two distinct graphs.

Finally, we moved to the "ABACADA non-means" signals, still using the same graphs for mapping. Figure 3 shows a MIDI pianoroll for the pieces generated with these signals, along with the composite BCI signal (attention and meditation eSense scales) [2]. Each piece is a rondo with the form *ABACADA*. In both pieces, the MIDI pianoroll shows clear repeated themes that are in sync with changes in the attention eSense meter. Increases in the attention level of the subject has a direct impact on thematic variations in the piece. Similarly, decreases in meditation (increased stress) also shapes the melody of the piece. The minor piece shows the impact of decreased meditation on the MIDI output. Both pieces share a similar temporal structure, even though the graphs used are entirely different. Thus, we have achieved a sort of separation of control between the graph (responsible for musical material) and the BCI input (responsible for temporal structure).

In previous iterations of this work evolutionary computation was used to search for good graphs. We have found that search is not necessary in this iteration, since the graph generator used to make initial graphs seems to give multiple "good" pieces out of every 5 generated. The pieces described above are using random seeds 1 and 5, where auditioning began at seed 0.

The pieces described here are available together with code, composite BCI signals, and a small collection of other pieces with low-numbered seeds, from
http://www.skynet.ie/~jmmcd/xg.html.

[2] Refer to the subject2_nonmeans_abacada mp3s available for download.

6. DISCUSSION & FUTURE WORK

We have succeeded in our initial steps: our representation can map BCI signals to music. It is responsive to changes in the signal, but not so responsive that changes in the signal lead to unrecognisable music. Static input signals lead to interesting musical patterns in a significant proportion of randomly-generated graphs, while the addition of variation in the input signals can lead to quite good "miniature" musical pieces. These statements are subjective, of course. One necessary step for future work is an objective validation.

The next phase of this project will then involve using the eSense meters in real-time. We will then have a feedback loop in our signal path: from the brain via the BCI, the graph, and the music, to the ear, and thence the brain.

Other issues to be investigated include: more fine-grained input data signals, rather than the two summary signals output by the eSense meters; the algorithm's sensitivity to parameters mentioned in Section 3.4; and the use of headsets for controlling an evolutionary search based on attention to multiple pieces of music in a population. The categorisation thresholds identified by [24] will be used to determine the success level of the generations, resulting in adaptive feedback composition. Multiple headsets will then allow collaborative composition.

7. REFERENCES

[1] O. Sacks, *Musicophilia: Tales of music and the brain*. New-York, Vintage Books, 2008.

[2] D. R. Hofstadter, *Le ton beau de Marot: In praise of the music of language*. Basic Books New York, 1997.

[3] A. W. Toga and J. C. Mazziotta, *Brain mapping: the methods*. Academic press, 2002.

[4] E. R. Miranda, W. L. Magee, J. J. Wilson, J. Eaton, and R. Palaniappan, "Brain-computer music interfacing (BCMI) from basic research to the real world of special needs," *Music and Medicine*, vol. 3, no. 3, pp. 134–140, 2011.

[5] A. Kirke, E. R. Miranda, and S. Nasuto, "Learning to make feelings: Expressive performance as a part of a machine learning tool for sound-based emotion therapy and control," in *Cross-Disciplinary Perspectives on Expressive Performance Workshop*, 2012.

[6] G. G. Molina, T. Tsoneva, and A. Nijholt, "Emotional Brain-Computer Interfaces," in *Affective Computing and Intelligent Interaction and Workshops*, 2009, pp. 1–9. [Online]. Available: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5349478&isnumber=5349257

[7] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, and T. M. Vaughan, "Brain–computer interfaces for communication and control," *Clinical neurophysiology*, vol. 113, no. 6, pp. 767–791, 2002.

[8] E. D. Adrian and B. H. C. Matthews, "The Berger rhythm: Potential changes from the occipital lobes in man," *Brain*, vol. 57, no. 4, 1934.

[9] E. R. Miranda, "Brain-computer music interface for composition and performance," *International Journal on Disability and Human Development*, vol. 5, no. 2, pp. 119–126, 2006.

[10] R. Teitelbaum, "In tune: Some early experiments in biofeedback music (1966-1974)," *Biofeedback and the Arts, Results of Early Experiments. Vancouver: Aesthetic Research Center of Canada Publications*, 1976.

[11] D. Rosenboom, *Biofeedback and the arts, results of early experiments*. Not Avail, 1976.

[12] ——, "The performing brain," *Computer Music Journal*, pp. 48–66, 1990.

[13] D. Williams, A. Kirke, E. R. Miranda, E. Roesch, I. Daly, and S. Nasuto, "Investigating affect in algorithmic composition systems," *Psychology of Music*, p. 0305735614543282, 2014.

[14] J. Eaton and E. Miranda, "Real-time notation using brainwave control," in *Sound and music computing conference (SMC)*, 2013.

[15] E. Miranda and A. Brouse, "Toward direct brain-computer musical interfaces," in *Proceedings of the 2005 conference on New interfaces for musical expression*. National University of Singapore, 2005, pp. 216–219.

[16] E. R. Miranda, K. Sharman, K. Kilborn, and A. Duncan, "On harnessing the electroencephalogram for the musical braincap," *Computer Music Journal*, vol. 27, no. 2, pp. 80–102, 2003.

[17] A. K. Hoover, M. P. Rosario, and K. O. Stanley, "Scaffolding for interactively evolving novel drum tracks for existing songs," in *Proceedings of EvoWorkshops*, ser. LNCS, vol. 4974. Springer, 2008, p. 412.

[18] A. K. Hoover and K. O. Stanley, "Exploiting functional relationships in musical composition," *Connection Science*, vol. 21, no. 2, pp. 227–251, 2009.

[19] J. McDermott and U.-M. O'Reilly, "An executable graph representation for evolutionary generative music," in *GECCO '11*, Dublin, 2011.

[20] J. Shao, J. McDermott, M. O'Neill, and A. Brabazon, "JIVE: A generative, interactive, virtual, evolutionary music system," in *EvoMUSART: Proceedings of EvoWorkshops*. Springer, 2010.

[21] D. Jackson, "A new, node-focused model for genetic programming," in *Proceedings of the 15th European Conference on Genetic Programming, EuroGP 2012*, ser. LNCS, A. Moraglio, S. Silva, K. Krawiec, P. Machado, and C. Cotta, Eds., vol. 7244. Malaga, Spain: Springer Verlag, 11-13 Apr. 2012, pp. 49–60.

[22] NeuroSky, "NeuroSky's eSense Meters and Detection of Mental State," NeuroSky, White Paper, September 2009.

[23] G. Rebolledo-Mendez, I. Dunwell, E. A. Martínez-Mirón, M. D. Vargas-Cerdán, S. de Freitas, F. Liarokapis, and A. R. García-Gaona, "Assessing neurosky's usability to detect attention levels in an assessment exercise," in *Human-Computer Interaction: New Trends*, ser. Lecture Notes in Computer Science, J. A. Jacko, Ed. Springer Berlin Heidelberg, 2009, vol. 5610, pp. 149–158. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-02574-7_17

[24] K. Crowley, A. Sliney, I. Pitt, and D. Murphy, "Evaluating a Brain-Computer Interface to Categorise Human Emotional Response," in *Advanced Learning Technologies (ICALT), 2010 IEEE 10th International Conference on*, July 2010, pp. 276 –278.

[25] ——, "Capturing and using emotion-based bci signals in experiments: how subject's effort can influence results," in *Proceedings of the 25th BCS Conference on Human-Computer Interaction*. British Computer Society, 2011, pp. 132–138.