2018-10-01

Composing with Biomemristors: Is Biocomputing the New Technology of Computer Music?

Miranda, Eduardo

10.1162/comj_a_00469
Computer Music Journal
Massachusetts Institute of Technology Press (MIT Press)

Music and Biocomputing: Is Music Biotech the New Computer Music?

Eduardo Reck Miranda*, Edward Braund* and Satvik Venkatesh*

*Interdisciplinary Centre for Computer Music Research (ICCMR)

Plymouth University

The House

Plymouth PL4 8AA

United Kingdom

{eduardo.miranda, edward.braund}@plymouth.ac.uk

satvik.venkatesh@students.plymouth.ac.uk

Abstract: Our research concerns the development of biocomputers using electronic components grown out of biological material. This paper reports the development of an unprecedented biological memristor and an approach to using such a biomemristors to build interactive generative music systems. The memristor is a relatively less well-known electronic component regarded as a resistor with memory. After an introduction to harnessing the *Physarum polycephalum* organism to implement biomemristors, the paper presents *PhyBox*: a biocomputer that uses four biomemristors to generate music interactively. The resistance of a biomemristor varies in function of voltage that has passed through it. Music input is represented in terms of voltage transitions and music output is encoded as measurements of current yielded by the system's memristive behaviour. An example of a musical composition using PhyBox is detailed. The paper concludes with a short discussion on how the combination between artificial

machines and biological organisms is paving the way for the development of new technologies for music based on living processors.

Introduction

In the late 1940s scientists at Australia's Council for Scientific and Industrial Research (CSIR) installed a loudspeaker on their Mk1 computer in order to track the progress of a program using sound. CSIR's Mk1 was one of only a handful of electronic computers in existence at the time. In 1951, Geoff Hill, a mathematician with a musical upbringing, programmed this machine to play back a melody (Doornbusch 2004). Then, in the late 1950s Lejaren Hiller and Leonard Isaacson composed *Illiac Suite for String Quartet* at the University of Illinois, USA, which is often cited as a pioneering piece of algorithmic computer music; that is, a piece involving materials composed by a computer. Its fourth movement, for instance, was generated with a probabilistic Markov chain (Hiller and Isaacson 1959).

The field of Computer Music has been progressing in tandem with the field of Computer Science ever since. A fair number of composers, such as Iannis Xenakis (1971), Max Mathews (Roads 1980), Pietro Grossi (Parolini 2017), Jean-Claude Risset (1985), and the pioneer of the so called *musique concrète*, Pierre Schaeffer (1971), to cite but five, have engaged with developments in computing at some point of their careers to develop their distinct approaches to music; see (Manning 2013) for a historical overview.

Whereas computing technology is omnipresent in the music industry today, future developments in computing will certainly continue to affect the way in which we create, perform and distribute music.

In addition to the availability of progressively more powerful and affordable equipment, advances in computer music technology have been characterized by the development of increasingly more user-friendly programming tools (Manzo 2011) and interfaces (Miranda and Wanderley 2006). These developments have enabled access to computer music technology to virtually anyone interested in using it, from amateurs to professional musicians. Our research, however, concerns the development of new kinds of processors at the core of computers and interfaces, and new approaches to making music engendered by such novel systems.

Scientists working in the emerging field of Unconventional Computing (UC) are developing methods to harness the immense parallelism and non-linearity of physical, chemical and biological systems to build new kinds of processors for electronic devices. Notable experiments have been developed to demonstrate the feasibility of building computers using reaction-diffusion chemical processors and biomolecular processors exploring the self-assembly properties of DNA (Adamatzky *et al.* 2003). The rationale behind these works is that natural chemical or biological agents would become components of the architecture rather than sources of inspiration to implement abstract models for software simulation. For instance, instead of modelling the functioning of neuronal networks for implementing deep learning algorithms, the UC approach endeavors to harness tissues of living brain cells to implement such algorithms. As an example of this approach for computer music, (Miranda *et al.* 2009) reported on the design of a sound synthesizer using brain cells cultured on a microelectrode chip.

Indeed, in the field of Computer Music there is a tradition of experimenting with scientific developments and emerging technologies. From countless examples of this, we cite but three notable works: (Choi 1994) experimented with chaotic oscillators for synthesizing sounds and

(Polotti and Evangelista 2001) presented one of a number of approaches developed by the computer music community to synthesize sounds based on fractal theory. Thirdly, (Weinberg and Driscoll 2006) introduced the intriguing concept of robotic musicianship inspired by the emerging developments in robotics at the time. Until very recently, however, developments put forward by the field of UC have been left largely unexploited by computer musicians. This is most probably so due to the heavy theoretical nature of research into UC, and the costly investment required to develop a laboratory and hire specially trained staff to build prototypes and conduct experiments. Nevertheless, research into UC for Music has been building momentum (Miranda 2017) as emerging research into building processors with the plasmodial slime mould *Physarum polycephalum* is proving to be an affordable way forward (Adamatzky 2016).

In following paragraphs the authors glance over *Physarum polycephalum* and briefly explain how it can be harnessed to build an electronic component referred to as the biomemristor. Next, they present the latest version of their *Physarum polycephalum* biomemirstor followed by an introduction to *PhyBox*, which is the latest incarnation of their interactive stand-alone musical biocomputer featuring four biomemirstors. Then, the paper demonstrates how PhyBox handles music and presents an example of a composition using the device.

An introduction to the basic research into UC with *Physarum polycephalum* that has been taking place at Plymouth University's Interdisciplinary Centre for Computer Music Research (ICCMR) can be found in (Miranda *et al.* 2017). In (Braund and Miranda 2017) the team reported on the implementation of ICCMR's first biomemirstor using this organism and more recently (Miranda and Braund 2017) described in great detail a method to build this component. This paper follows from the group's previous publications with an introduction to PhyBox

accompanied with a thorough description of an approach to processes music with it. Moreover, it also details for the first time how the system has been put into practice to compose an unprecedented interactive piece for piano and percussion entitled *Biocomputer Rhythms*.

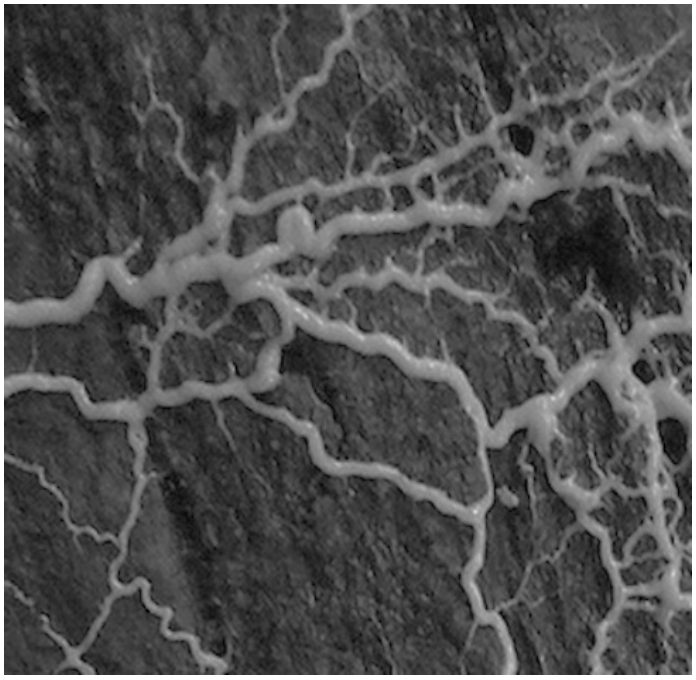
Physarum polycephalum

Physarum polycephalum, henceforth referred to as *P. polycephalum*, is a slime mold found in cool, moist and dark environments; e.g., in the woods, on decaying leaves and tree bark (Figure 1a). It is a single eukaryotic cell with many heads; hence the term 'polycephalum'. It is typically yellow in color and visible to the naked eye. This organism feeds through a process called phagocytosis: it coats its food in enzymes, which allow for specific nutrients to be ingested, leaving behind a mass of unwanted material. In the laboratory, we normally use oat flakes to feed it (Figure 1b).

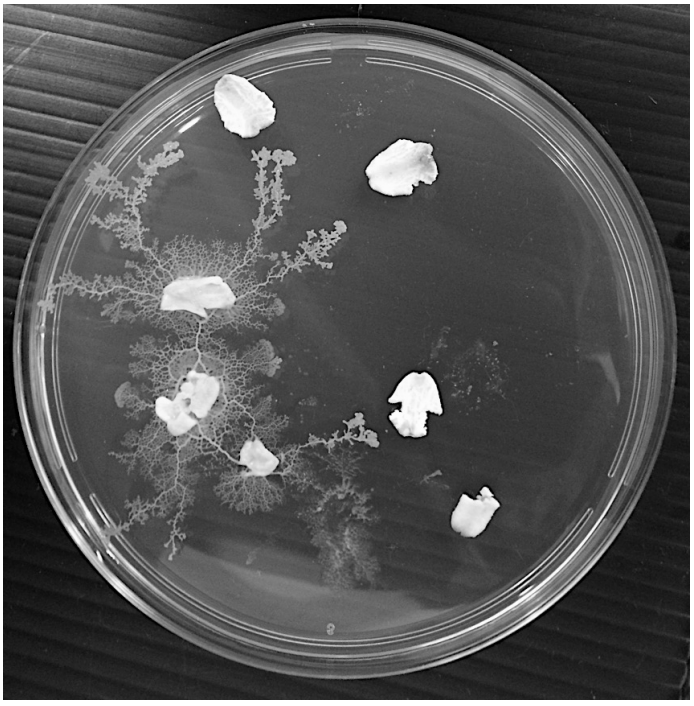
P. polycephalum exhibits a complex lifecycle (Howard 1931), but the point of interest here is its plasmodium stage, which is when the organism actively searches for food. As it does so, it grows a network of protoplasmic tubes that rhythmically contract and expand producing shuttle streaming of its internal fluid.

The organism is straightforward to culture in Petri dishes and it is relatively easy to prompt it to grow specific topologies of protoplasmic tubes by placing food and repellents (e.g., salt) strategically on the dish. The ability to manipulate its growth patterns has underpinned the early stages of research into building *P. polycephalum*-based machines to realize tasks deemed as computational: e.g., the organism was prompted to find the shortest path to a target destination

through a maze (Adamatzky 2016) and solve the classic combinatorial optimization Steiner tree problem (Caleffi *et al.* 2015).



(a)



(b)

Figure 1. (a) P. polycephalum can be found on decaying leaves and tree bark. (b) A culture in a Petri dish.

The focus of our research is the electrical behavior of *P. polycephalum*. Its intracellular activity produces fluctuating levels of electrical potential as pressure within the cell changes. Typically, this is in the range of +/- 50mV, displaying oscillations at periods between 50s and 200s, with amplitudes ranging from 5mV to 10mV (Meyer and Stocking 1970). Interestingly, electrical current can be relayed through its protoplasmic tube, and this prompts it to behave like an electronic component, referred to as the *memristor*.

The memristor

The memristor is the relatively less well-known fourth fundamental passive electronic circuit element. The other three are the resistor, the capacitor and the inductor. The word 'memristor' is a contraction of two words, 'memory' and 'resistor'. It can be thought of as a resistor with memory, because its resistance depends on the history of previous inputs (Trefzer 2017). This element was outlined by Leon Chua in 1971 when he proved theoretically that its behavior could not be simulated by any of the other three fundamental circuit elements, or combinations thereof (Chua 1971). However, the theory was not connected to a physical device until 2008 (Strukov *et al.* 2008).

The memristor is an element with two terminals. The magnetic flux between the terminals at any given time depends on the charge that has passed though it in the past. In other words, the memristor alters its resistance as a function of the previous input voltage and the amount of

time that this voltage was applied; this property is referred to as hysteresis. When the application of the voltage stops the element retains its most recent resistance state. Mathematically, we can use a state-dependent Ohm's laws to define memristance M as the element's resistance R to a given charge q as follows:

$$M = R(q) = \frac{d\delta(q)}{dq}$$

where q is charge, δ is flux and d is the derivative, denoting change in flux with respect to change in charge. In contrast to the other three fundamental elements, a memristor can exhibit nonlinear behavior. If the value of q is constant, then, over time, the memristor would maintain a linear relationship between voltage and current, similarly to a resistor. However, if q is variable (e.g. an AC voltage), then this relationship becomes nonlinear.

Figure 2 shows the memristor's current-voltage characteristic hysteresis curve, where a high or a low resistance pathway is followed according to whether the voltage is increasing or decreasing. (Chua 1971) established that if a memeristor produces a symmetric figure of eight curve with the center intersection at zero voltage and zero current, then it is considered as an 'ideal memristor'. The magnitude of hysteresis curve lobe size changes as a function of the frequency of the input voltage and the memristive system's response time.

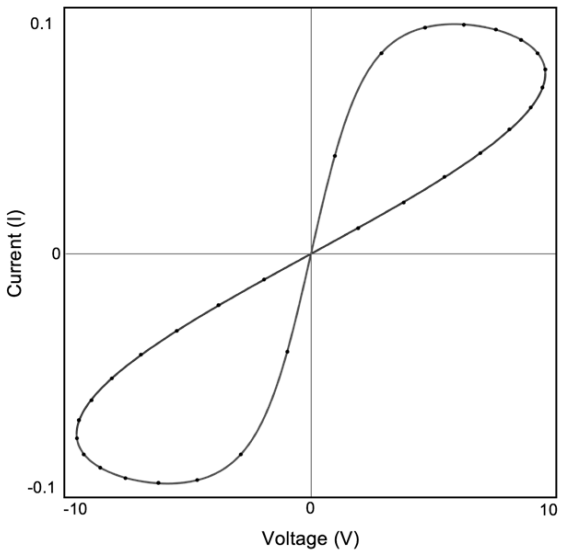


Figure 2. The ideal memristor's current-voltage characteristic hysteresis curve.

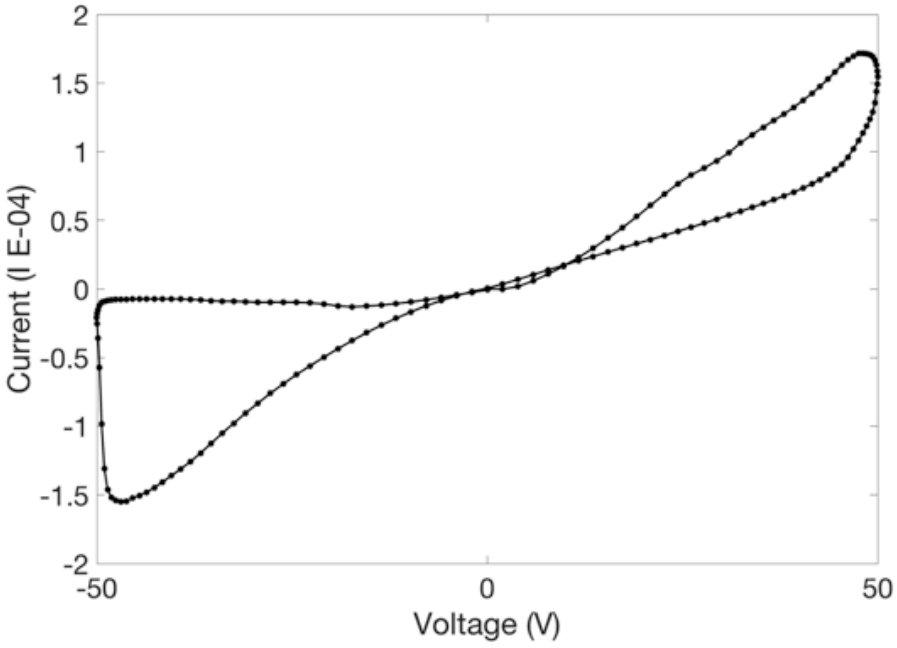
The memristor is exciting for computer scientists because its behavior has been found to be comparable to the behavior of biological neurons and certain processes in the brain, which is paving the way for the development of brain-like processors. For instance, sudden changes in voltage prompt the memristor to produce spikes of current (Figure 12), which is a behavior comparable to the spikes produced by neurons (Versace and Chandler 2010).

Although interest in the memristor is thriving, it has not been widely deployed to date. On the one hand, the electronics industry still is in the process of developing a robust cost-effective memristor chip for manufacturing. On the other hand, revolutionary new appliances using this component need to be invented to create demand for memristors on an industrial scale. In the meantime, the discovery that *P. polycephalum* can be harnessed to act as a memristor is providing an alternative route for making memristors, which is to grown them out of biological material. A detailed introduction to the memristor is available in (Trefzer 2017) and an in-depth theoretical explanation can be found in (Chua 2015).

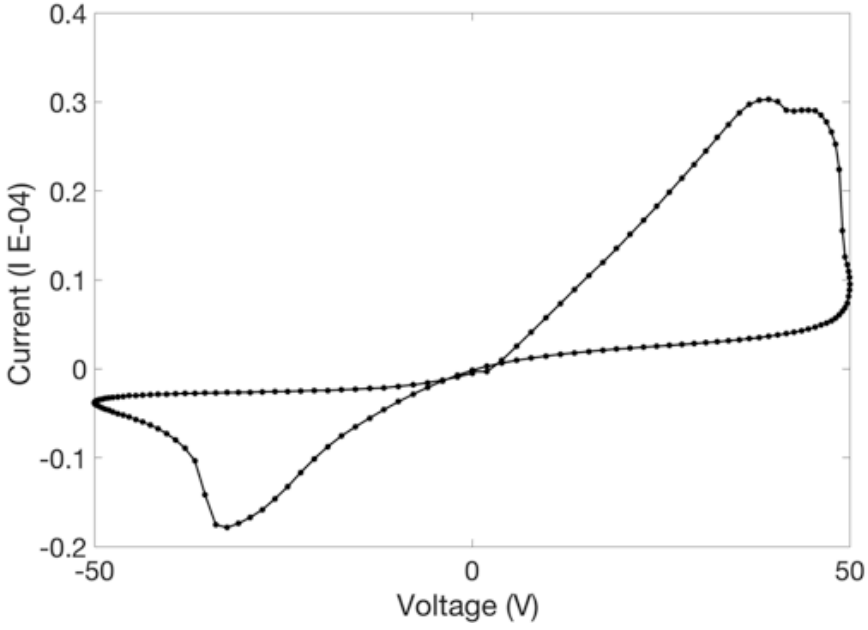
Physarum polycephalum biomemristors

A number of experiments have shown that *P. polycephalum* behaves like a memristor (Pershin *et al.* 2009; Gale *et al.* 2013). The ICCMR team demonstrated that *P. polycephalum* produces current-voltage hysteresis curves in response to systematic application of varying AC voltage, which are comparable with the memristor's curve shown in Figure 2. These current-voltage profiles were measured using discretized sinusoid AC voltage waveforms of 160 steps ranging from +/-0.05V to +/-50V, with step dwell times ranging from 0.5s to 2.5s. Dwell time is the duration of a voltage application step before moving onto the next step; for instance, if dwell time is set equal to 2.0s, then 160 cycles would take 320s to complete. An ammeter made instantaneous current measurements at each voltage step. Interestingly, we observed that the curve's shape varied dynamically, but remained consistent with the memristor's characteristic curve (Figure 3). This anomaly could be due to external factors that influence the organism, like humidity, temperature, light and electrical history. According to Chua's hypothesis, the ideal memristor consistently produces the same hysteresis curve. However, these minor variations generated by *P. polycephalum* can be an advantage for building computer-aided generative music systems. For instance, we envisage developing systems with a controllable coefficient of nonlinearity, which would enable us to specify the degree of variation we would expect the system to produce in relation to input data. This will become clearer when we introduce below of the methods that we developed for music processing with biomemristors.

The encouraging results from the experiments mentioned above prompted us to begin research aimed at the design and implementation of practical biomemristors with a view on building intelligent interactive music systems with them.



(a)



(b)

Figure 3. Two examples of current-voltage curves obtained from experiments with P.polycephalum conducted by the authors.

In order to conduct this research, we maintain a *P. polycephalum* farm that adapts techniques from (Adamatzky 2010). The first prototypes were implemented on Petri dishes retrofitted with electrodes made with circles of tinned copper wire filled with non-nutrient agar (Figure 4). This enabled them to grow a protoplasmic tube connecting the two electrodes, which would effectively create the biomemristor. In order to prompt the organism to lay down the required protoplasmic tube, we positioned a *P. polycephalum*-colonized oat flake surgically extracted from the farm on one of the electrodes, and a fresh oat flake on the other. This arrangement influences the organism to grow towards the fresh oat, resulting in a protoplasmic tube linking the two electrodes.

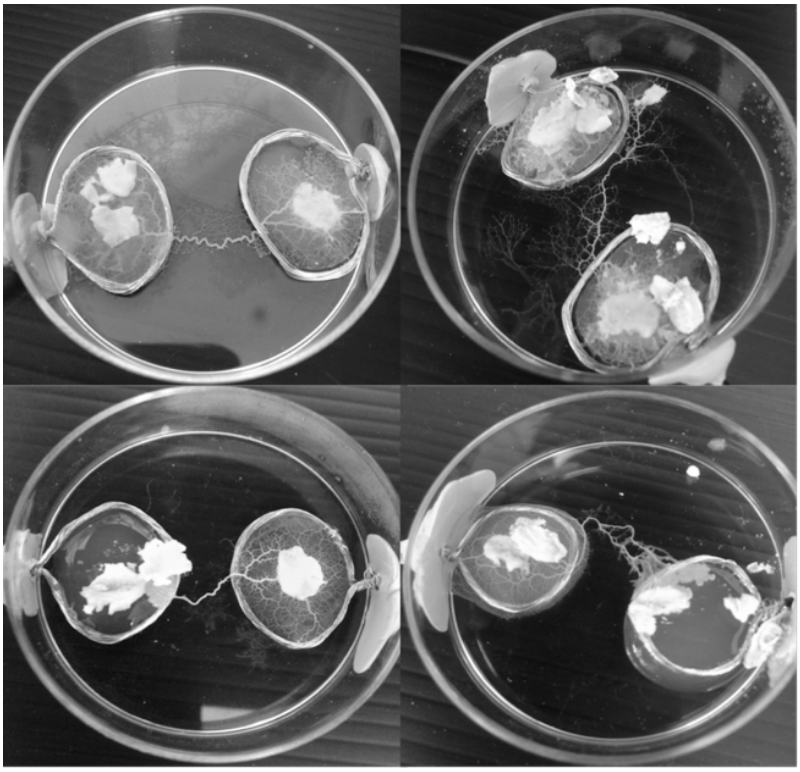


Figure 4. Photographs of four biomemristor implemented in a 60mm Petri dish.

The process of building the prototypes highlighted a number of pitfalls that needed to be addressed in order to develop practical components to be tested in realistic applications. For

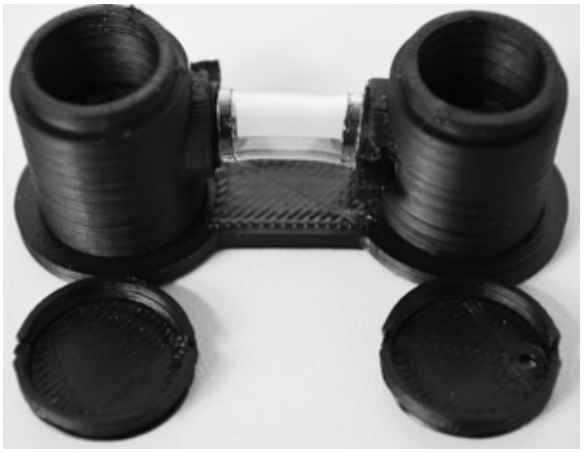
instance, fitting Petri dishes with the necessary parts was complicated and time-consuming. Plus, it took circa 30 hours to grow a protoplasmic tube in a Petri dish. This was most probably so because growth conditions were not delineated satisfactorily; for instance, within the Petri dishes the organism has a number of different propagation trajectories and grows in unpredictable ways. As a result, components had a high degree of morphological variation from one another and memristive observations also differed vastly between them. Another drawback was the component's lifespan: a protoplasmic tube remained functional for only approximately 40 hours on average, and this was not under constant usage. Moreover, the components were rather fragile, rendering them unpractical for usage outside controlled laboratory conditions.

In order to address the above-mentioned problems, a receptacle was developed to culture the organism in a more controlled way than before. The receptacle, which is fabricated using 3D printing, encapsulates the organism into a stable environment, which delineates a clear constrained propagation trajectory.

The receptacle consists of chambers connected via a tube (Figure 5a). The chambers can accommodate 1.5ml of agar to achieve a favourable level of humidity. To delineate the growth of the protoplasmic tube, we fabricated the chambers with high impact polystyrene (HIPS) as this substance is a repellent for *P. polycephalum* (Gotoh and Kuroda 1982). Consequently, the plasmodium will be discouraged from growing on the walls of the chamber and encouraged to propagate across the linking tube to the other chamber, laying down the desired protoplasmic tube (Figure 5b). We learned the hard way that the plasmodium does not propagate well over bare metals. Thus, we decided to avoid using metal electrodes in favour of more a biocompatible material. To this end we used a biocompatible conductive polylactic acid (PLA) 3D printing material to embed the electrodes in the receptacle instead. PLA is Food and

Drug Administration certified and is widely used in the biomedical applications (Gupta *et al.* 2007). Using this material, we printed two collars that slotted into the chambers. Each collar was designed with an electrical contact point and a rim to attach the linking tube between the chambers (Figure 5). For the linking tube, we used off-the-shelf medical grade polyvinyl chloride (PVC) tubing, which is available in a variety of inner and outer tube dimensions; we adopted tubing that had a 4mm inner diameter and mm outer diameter.

The new 3D printed receptacle enabled for standardization of growth conditions and electrical properties, and increased the life span of the component. Growth time decreased to less than ten hours with the receptacle, and the samples always grew the required protoplasmic tube with no exceptions. Delivering voltage to the organism does reduce its life spam, but the receptacle alleviated this problem considerably. We build ten components under identical conditions and fed them with continuous voltage. We measured them once a day until they presented no memristive curves. Seven of them maintained their memristance for an average of seven days, with three samples doing so for as long as fourteen days.


(a)

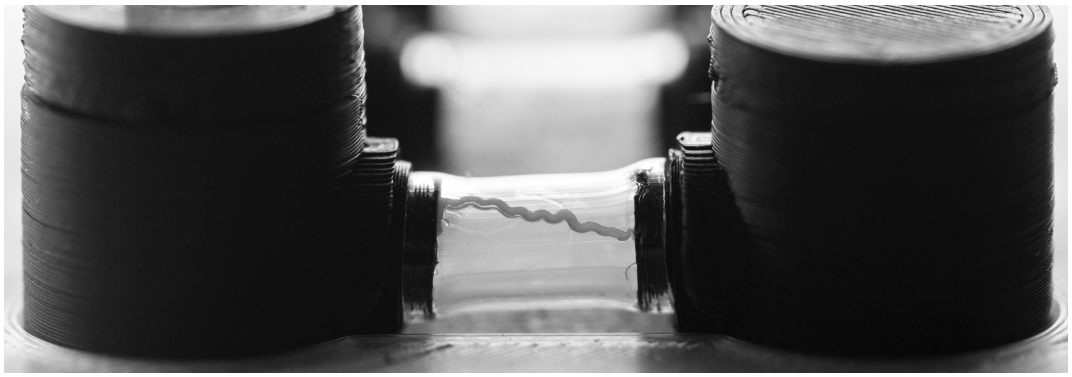
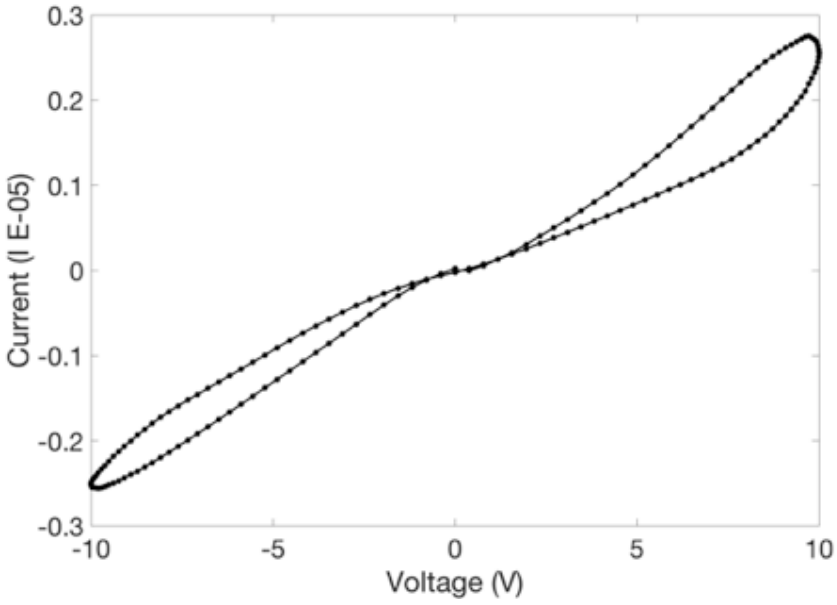
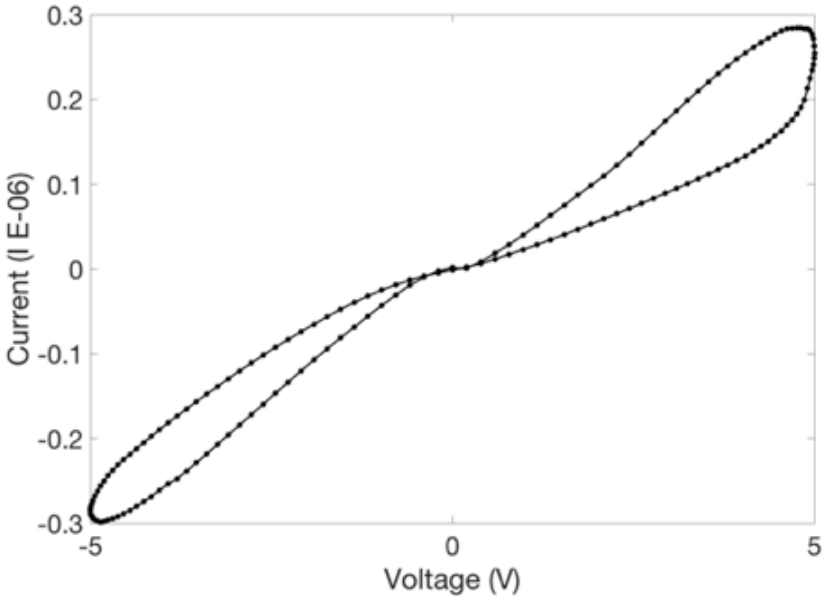

(b)

Figure 5. (a) An empty receptacle. It is size is 45mm. (b) A receptacle with a cultured organism. Note the protoplasmic tube linking the two chambers.

Conductive 3D printing filament provided an efficient way to integrate the component in an electronic circuit. And more, the component yielded current-voltage curves that were more symmetrical than those obtained with the previous Petri dish setup, and with consistent lobe sizes and pinch locations (Figure 6). For more details on the benefits of the new receptacle-based component and testing please refer to (Braund and Miranda 2017) and instructions for building one can be found in (Miranda and Braund 2017). The encouraging performance of the improved design enabled us to take the biomemristor out of the laboratory and test it in the real world. Below we introduce PhyBox, an interactive musical biocomputer with four biomemristors, followed by an explanation of how it handles music.



(a)



(b)

Figure 6. Two examples of current-voltage curves measured with samples in the receptacles.

PhyBox

A biomemristor handles input data in terms of voltages and outputs data in terms of current. At present, PhyBox holds four biomemristors, each of which requires a voltage source for input and an ammeter to measure current for output. The core of PhyBox hybrid wetware-hardware architecture is a custom-built microcontroller furnished with DAC and ADC breakout boards programmed to handle input voltage impulses for the biomemristors and output readings of current. A MIDI interface is also embedded for music input and output.

Firstly, the system splits the music input into four streams of data, one for each biomemristor: pitch, loudness, inter-onset interval and duration. Next, each stream is converted into a sequence of voltage impulses for inputting to the respective biomemristor. Then, measurements of current from the biomemristors are converted back into the respective streams of data that will constitute the resulting music output.

The PhyBox circuitry is cased in a robust and portable 3D printed box. The lids of the box incorporate sockets that allow for the biomemristors to be easily clipped in and out of the circuit, providing means of quick component replacement (Figure 7). The device is powered via a USB port and a touch display can be plugged in for operation. Although PhyBox operates as a stand-alone device, the USB port enables communication with a standard computer if required; e.g., for uploading and downloading software or MIDI data. Although PhyBox was built to handle music data encoded as a MIDI signal, it is equally possible to work with audio for input and raw measurements of current for output. In this case the device needs to be connected via USB to an external computer to handle audio and render current measurements into music in a format other than MIDI.

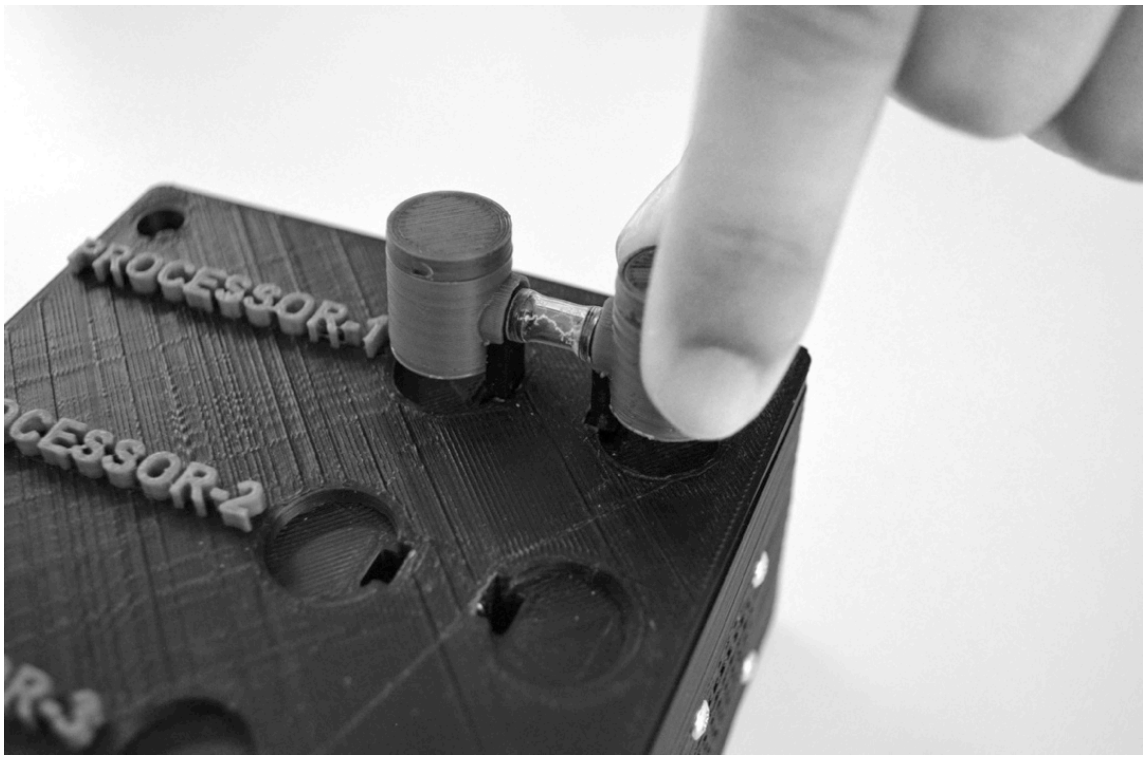


Figure 7. The PhyBox and one biomeristor being clipped in.

Music processing

The magnitude of a spike is directly related to the difference between the voltage of the incoming impulse and the biomemristor's present voltage engendered by the previously input. The greater this difference, the higher the magnitude of the spike (Figure 12). Furthermore, the greater the voltage difference, the longer the biomemristor takes to spike. These are the core properties of the biomemristors that were harnessed by the methods that we have been developing to process music, one of which is explained in this section.

For the sake of clarity, the explanation below focuses only on pitch, and considers a case whereby it reads a MIDI file encoding a short musical excerpt first and then generates a musical

response, after the whole excerpt has been converted into voltage impulses. However, the system can also work on the fly for interactive music scenarios, as it will be shown when we introduce the musical composition example later on.

As the music stream is processed the system generates voltage impulses ϕ and each pitch is stored with its respective number of occurrences so far. An interim voltage value V in the range of 0V to 10V is calculated as follows:

$$V = 10 - \left(\left(\frac{10}{N}\right) \times n\right)$$

where, N is the total number of processed events so far and n is the number of times the present event has occurred up to this point. Then, if the present event has occurred more frequently than the previous one, the value of the impulse ϕ^t is calculated by increasing the positivity or negativity of the previous impulse ϕ^{t-1} by the magnitude of voltage V, depending on the polarity of ϕ^{t-1}. Otherwise, ϕ^t is calculated by decreasing the positivity or negativity of the previous impulse ϕ^{t-1} by the value of V. Note that the increase or decrease occurs here in terms of voltage magnitude, which could be either positive or negative. As an example, let us consider the excerpt from J. S. Bach's *Gavotte en rondeau* shown in Figure 8.

The first event is note B4, or note MIDI number 71. In this case $V = 10 - \left((10/1) \times 1\right) = 0$V, and as this is the first event, then $\phi^1 = 0$.

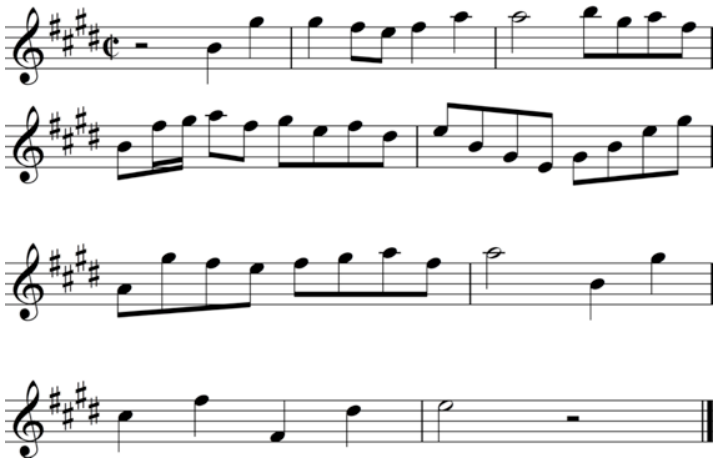


Figure 8. *Excerpt from J. S. Bach's 'Gavotte en rondeau'.*

Next, comes the second event, which is MIDI note 80. The voltage for this note is calculated as $V = 10 - \big((10/2) \times 1\big) = 5$V. As this is only the second event and the magnitude of the previous impulse is neither positive nor negative, the system arbitrarily makes it as a positive impulse: $\phi^2 = V$ (Table 1).

Event	Note	n	V	ϕ
1	71	1	0.00	0.00
2	80	1	5.00	+5.00

Table 1: Voltage impulses assigned to two events.

Then, comes the third note, which is also MIDI note 80. As this note occurred more times than note 71, it respective impulse is calculated by increasing the positive value of the previous impulse by the present voltage value: $\phi = 5.00 + 3.33 = 8.33$ (Table 2).

Event	Note	n	V	ϕ
1	71	1	10.00	0.00
2	80	1	5.00	+5.00
3	80	2	3.33	+8.33

Table 2. Voltage impulses assigned to three events.

Next in the sequence is MIDI note 78, which occurred less frequently than the previous notes. In this case, the impulse is calculated by decreasing the positive value of the previous impulse by the voltage value for note 78, that is: $\phi = 8.33 - 7.50 = 0.83$. The fifth note is 76, which occurred the same number of times as the previous one. Therefore $V = 10 - ((10/5) \times 1) = 8.00$ and the impulse is calculated by decreasing the positive magnitude of the previous impulse, which brings it down to a negative value: $\phi = 0.33 - 8.00 = -7.17$ (Table 3).

Event	Note	n	V	ϕ
1	71	1	0.00	0.00
2	80	1	5.00	+5.00
3	80	2	3.33	+8.33
4	78	1	7.50	+0.83
5	76	1	8.00	-7.17

Table 3. Voltage impulses assigned to four events.

Table A1 in the Appendix shows the complete list of input notes of the excerpt in Figure 10, their number of occurrences $\boldsymbol{n}$, respective interim voltages $\boldsymbol{V}$ and impulse values $\boldsymbol{\phi}$. The resulting impulse sequence is plotted in Figure 9.

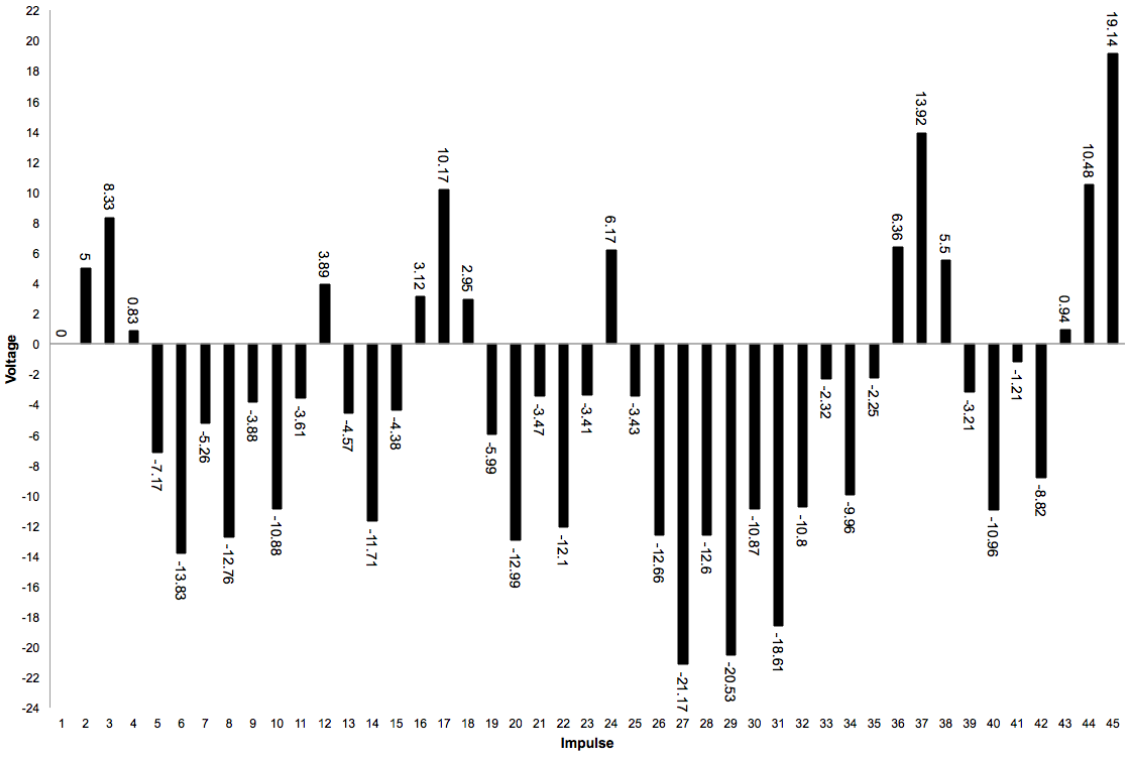


Figure 9. Voltage impulse sequence for J. S. Bach's 'Gavotte en rondeau'.

While the system calculates the values of the impulses, it also builds a transition table of *inverted percentages* of note occurrences (Table 4). As the musical input is processed, the system dynamically calculates percentages of transitions between two events. These percentages are subsequently inverted to make smaller values denote greater occurrence of a certain transition from one given note to another, and vice-versa. This aligns the musical transitions with the behavior of the biomemristor: it produces low memristance as the voltage increases and high

memristance as the voltage decreases. Therefore, small changes from one voltage impulse to another encode more frequent transitions, whereas large changes encode less frequent ones.

	64	**66**	**68**	**69**	**71**	**73**	**75**	**76**	**78**	**80**	**81**	**83**
64			0.0									
66							0.0					
68	50.0				50.0							
69										0.0		
71			80.0					80.0	80.0	60.0		
73									0.0			
75								0.0				
76					80.0				40.0	80.0		
78		90.0			90.0		90.0	80.0		70.0	80.0	
80				88.8		88.8		88.8	77.7	88.8	66.6	
81					83.3				50.0		83.3	83.3
83										0.0		

Table 4. Table of inverted percentages of transitions from notes listed on the vertical axis to the ones listed on the horizontal axis.

The voltage impulses are then applied one at a time to the respective biomemristor, the corresponding current is measured, and this value is subsequently used to generate a note for output.

In order to translate from measurements of current to MIDI note numbers, each current reading I^t is compared against its predecessor's I^{t-1} to calculate an absolute change rate value ΔI, as follows:

$$\Delta I = \left| \left(\frac{(I^t - I^{t-1})}{I^{t-1}} \right) \times 100 \right|$$

Then, the system selects the option in the transition matrix whose inverted percentage value is the closest possible to the value of ΔI. To start with, the system considers the first note of the input music, which is equal to 71. For example, the current reading for the first impulse $\phi^1 = 0$ (corresponding to the first input note 71) is $I^1 = 0.0252 \times 10^{-4}$. As there is no predecessor value for the ΔI equation, the system establishes that $\Delta I^1 = 0.0$ and picks note 80 from Table 4 because note 71's inverse probability value of 60.0 is the closest to $\Delta I^1 = 0.0$. Next, for the second input note 80, $\phi^2 = 5.0$ yields $I^2 = 0.1961 \times 10^{-4}$, therefore $\Delta I^2 = 678.17$. In this case, the closest inverted percentage is 88.8. From the four choices available the system picks note 69. For the third input note, also 80, $\phi^3 = 8.33$ produced $I^3 = 0.2053 \times 10^{-4}$ and $\Delta I^3 = 4.69$. Therefore, the system picked note 81, whose transition has the lowest inverse probability value equal to 66.6. So far the system generated notes 80, 69 and 81 as responses to notes 71, 80 and 80, respectively. The complete list of currents, change rate values and output notes are given in Table A1 in the Appendix. A plot of the currents yielded by the biomemristor is shown in Figure 10 and the resulting notes in standard musical notation is shown in Figure 11. Obviously, the temporal structural of the Bach input has been discarded in Figure 11 because the examples focused only on pitch processing.

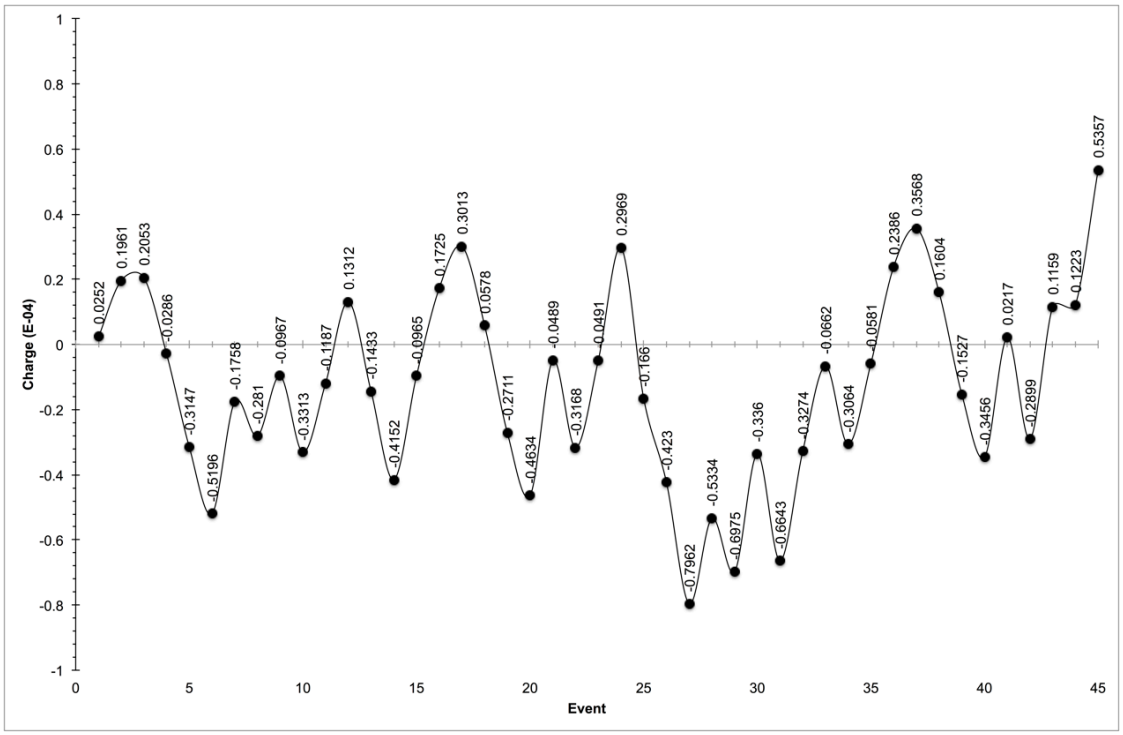


Figure 10. The currents yielded by the biomemristor.

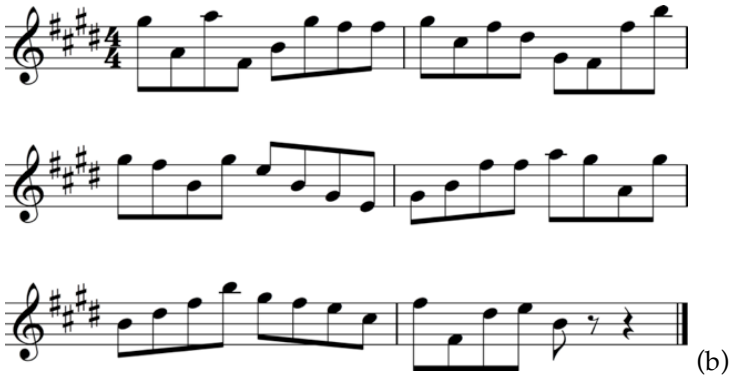


Figure 11. Music output from the system. Note that only pitches were processed by the system, hence the temporal structure of the Bach's input is lost.

As we have seen earlier, upon the application of a voltage impulse, the biomemristor produces a spike in its running current, and goes back to exhibiting a steady current flow. Current readings can be controlled with two adjustable parameters: dwell time and offset

percentage. As mentioned earlier, step dwell time in milliseconds defines how long the impulse voltage is applied to the biomemristor and a measurement of percentage specifies when to measure the current. For instance if the dwell time is 2s and offset percentage is 25% then the current is measured 0.5s after the voltage has been applied. Figure 12 shows a biomemristor's response to a change in voltage. The shorter the dwell time and offset, the less time the component has to respond to the change. In practice, the closer to the spike the higher the current value, and the more variation we would obtain in the musical output. The dwell time for the Bach example above was 2s and the offset percentage was 50%; that is, current was measured 1s after voltage onset.

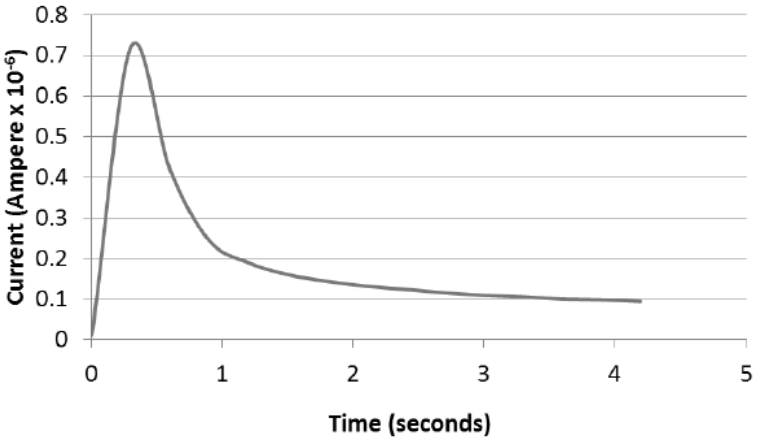


Figure 12. A sudden change in voltage produces a peak in current.

Earlier in this paper we suggested that the degree of variation of the musical output in relation to the input could be made controllable if a way to handle the hysteresis of the system is found. A biomemristor with a different current-voltage profile from the one used for the above example would have produced different current readings and consequently a distinct musical output from the one shown in Figure 11. Research is well under way to develop effective methods to vary the hysteresis behavior of the component and respective new music mapping procedures. For instance, we have conducted preliminary experiments whereby we were able to

force *P. polycephalum* to behave like an ordinary resistor by pipetting a chemical agent on it. In this case current flow through the component remained constant for a given voltage. Then, we adapted the above mapping scheme to produce music output that is identical to the input when the biocomponent acted as a resistor and variations of the input when it acted as a memristor. There is much research still to be done in order to a) engender behaviors ranging from that of a resistor to that of a highly 'non-ideal memristor' (Chua 1971) and b) design methods that harness these phenomena to process music.

A composition example: *Biocomputer Rhythms*

Biocomputer Rhythms, for piano and percussion, was composed by the first author in tandem with the development of PhyBox hardware and the music processing method introduced above. It is important to note that ideas behind the musical composition informed the design of the technology and vice-versa. It is our research modus operandi to take research out of the laboratory for testing in the real world at the development stage. As a matter of fact, the first performance of the piece still banked on an unfinished version of PhyBox, with voltages and currents handled by Keithley 230 voltage sources and Keithley 617 electrometers (Miranda 2016a; 2016b).

The setting for *Biocomputer Rhythms* is as follows: a microphone takes audio from the piano, which is subsequently analyzed to calculate pitch, loudness, inter-onset intervals and duration of notes, respectively. Then, this information is input to the biomemristors, as explained above. Current readings are converted into sequences of MIDI notes, which are relayed to an adapted version of McPherson's Magnetic Resonator Piano (McPherson 2010), consisting of electromagnetic actuators that play the piano (i.e., the same instrument as the one played by the

pianist during the performance) and percussion instruments. A set of 24 electromagnets is positioned inside the piano to vibrate the strings of 24 notes. In addition, six electromagnets are allocated to vibrate six percussion instruments: a tam-tam, a snare drum, a suspended cymbal, tubular bells, a metal washer inside a colander, and a disposable foil tray.

The most common weaknesses of algorithmic music composition systems in general is their lack of capability to take into account musical form; by musical form we mean the overall structure of a musical composition conveyed by the grouping and disposition of musical materials delineating distinct sections of the piece. Hence, after a while the output from such systems tends to becomes repetitive and less appealing to listeners. If the truth be told, musical form is the aspect of a generative music system's output that is most often manually tweaked. There have been a number of approaches to address this shortcoming; for instance, by processing music at various hierarchical levels, such as note sequences, motifs, sections, stanzas and movements, respectively. An approach that is popular in the Artificial Intelligence and music literature follows the work of Fred Lerdahl and Ray Jackendoff (1983) into music analysis inspired by principles of linguistics. However, such approaches are often biased to particular types of music, which most often are tonal classical music, jazz and simple popular musical forms.

Although PhyBox could be used as an automatic music generator on its own right, we are not developing this research with this application in mind. We consider PhyBox as an interactive music agent. Therefore, we are not concerned with tackling the issue of musical form automatically here. We prefer to regard this as a matter for composition design instead. Having said that, as the research develops we might be tempted to embed in the system's design the experience that we will have eventually amassed from designing compositions with it.

The composer designed the piece as a sequence of 14 scenes, which can be played in any order. The important thing here is that the scenes are different from each other. Each scene, presented to the performer as a one-page long score, consists of: (a) the music to be played, (b) specification of listening windows and (c) settings for the electromagnetic actuators system, or EAS (Figure 13).

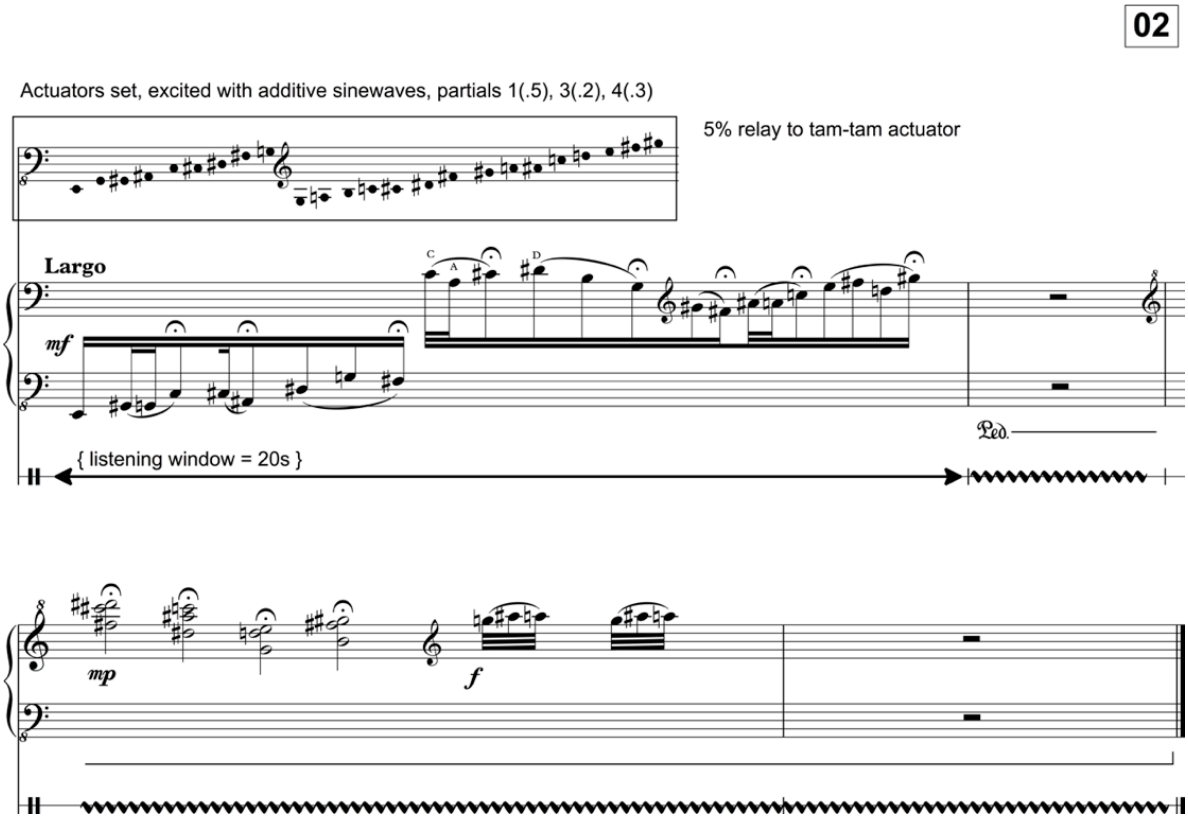


Figure 13. 'Biocomputer Rhythms', scene number 2.

The motivation for preparing the piano with the electromagnets stems from the composer's desire give the piano a dual identity: one characterized by the standard piano sounds, which are produced by activating the hammers to strike the strings, and the other characterized the somewhat other-worldly sounds, which are produced by vibrating the strings by means of the electromagnets. The percussion instruments were included to add more variety to the available

palette of sounds to compose with. In the end, they inadvertently added a freakish aspect to the piece: they looked as though they were played by an invisible, or unembodied, entity on stage.

Listening windows, indicated on the score below the piano part, specify which portions of music will be input into PhyBox. For instance, in scene number two, shown in Figure 13, PhyBox is set into listening mode for the first 20s, which is the time the performer is given to play the first bar. As the pianist plays the first bar, the notes are input to the system. Then, after 20s the system shuts the listening window and enters into generative mode, indicated by a wiggly line. The system remains in generative mode until a new listening window is opened. Nothing is input into the biomemristors while PhyBox is in generative mode. However, the system can listen to itself. That is, once the processing of the original sequence played by the pianist has finished it will consider the most recent generated note as a reference to produce the next input voltage for the biomemristors. The tables of inverted percentages of transitions are updated with the newly generated note, and so on. The performer opens and shuts listening windows via PhyBox's touch screen interface. For the premiere of the piece however, this was done via an iPad that also controlled settings for the EAS.

Each scene deploys different sub-sets of electromagnetic actuators. For instance, note that in scene number twelve, shown in Figure 14, only a sub-set of eight notes is available for PhyBox, whereas all of the 24 notes are available in scene number two; this is represented inside a box above the piano part. If the biomemristors produce a note that is not available, then system plays the nearest available note in the respective sub-set.

The EAS activates the electromagnets by means of audio. In fact, each electromagnet could be regarded as a small loudspeaker without the vibrating cone. For *Biocomputer Rhythms*, the

electromagnets are activated with a sound wave comprising three partials, calculated from the pitch of the respective note to be played. In scene number two, for example, the waveform is composed of the fundamental frequency, the third and the fourth partials, respectively; at the top of the score, the relative amplitudes are indicated in parenthesis next to the partial number. The waveform of the activating sound varies from scene to scene, producing slight variations on the other-worldly timbres yielded by the vibrating strings of the piano.

Finally, at each scene the EAS is programmed with a certain probability of activating percussion instruments instead of notes of the given sub-set. For instance, there is a 60% chance in scene 12 that a note might be played by the tubular bells or foil tray instead of the piano; this is indicated on the right-hand side of the box above the piano part.

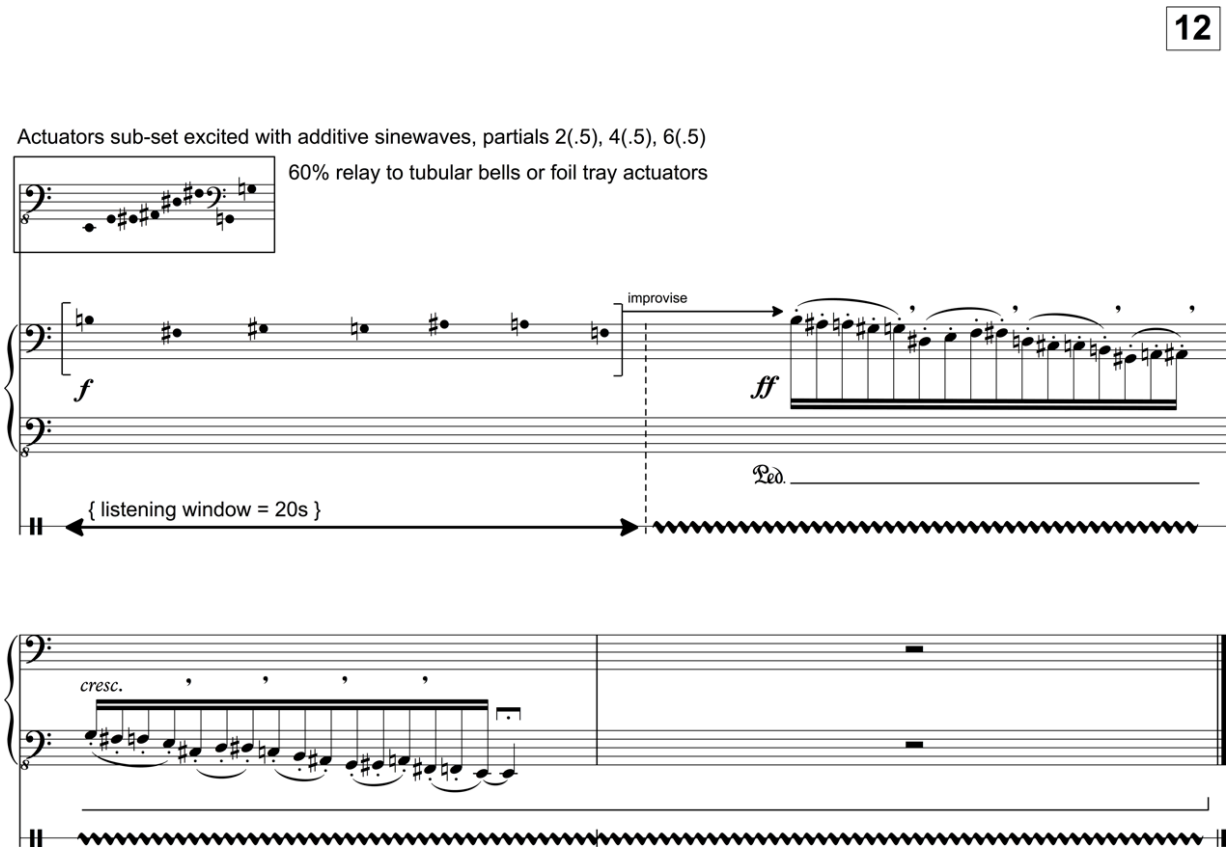


Figure 14. 'Biocomputer Rhythms', scene 12.

The piece was premiered on February 2016 at Peninsula Arts Contemporary Music Festival, Plymouth, by the composer himself on the piano with technical assistance of the second author; a video recording pf this concert is available: (Miranda 2016a). Even though the composer had been working with *P. polycephalum* for a few years, the premiere of this piece felt strangely spooky to him. The fact that he was interacting with a living machine on stage created an aura of mystery and the audience seemed in awe of the piece. They flocked to the stage after the performance to ask questions and check the setup for themselves. Not at all surprisingly, the majority of the questions were about the technology rather than the music, which is the thing that we really wanted to talk about with the audience. However, we believe that this is part of the research process: as bio-systems such as PhyBox become more widely used, their novelty will wear off and focus on the music they actually engender should then prevail.

Concluding discussion

This paper introduced an approach to exploring the potential of biocomputing for music using *P. polycephalum*. There are biological systems other than *P. polycephalum* that exhibit memristive properties. However, they have significant constraints that limit their viability to develop an actual electronic component. The plant *Aloe vera* is a known example (Volkov *et al.* 2014), but it requires controlled light conditions to grow, has a very much limited lifespan, and would be difficult to integrate into a circuit.

P. polycephalum is an excellent organism for our research because of the simplicity of its morphology and its ability to grow on most substrates, forming networks of wire-like veins. The cell's morphology can be delineated straightforwardly to conform to a conventional electrical

scheme. What is lacking is a better understanding of these wire-like veins, their growth process and the cellular processes that contribute to their memristive properties. Much research is needed to understand and control the processes operating within *P. polycephalum* that give rise to its memristive properties.

We have been able to grow components that remain active for seven days, but we have not monitored how their memristance changes as time progresses, in particular toward the end of their life. We have not used them for more than 2 hours and each time we conduct an experiment or a performance we deploy new units. It would be interesting, however, to understand how memristance may change with time, as this could most certainly be explored musically. And indeed, it is the fact their behavior change over time and that they have a limited life span that makes it exciting to work with biomemristors.

At the beginning of our research journey we were concerned with establishing relationships between specific properties of *P. polycephalum* and music. For instance, (Miranda *et al.* 2011) reported an experiment whereby data related to the electrical activity produced by the plasmodium during foraging and its spatial-temporal behavior were converted into sound. However, once we learned about the memristive properties of the organisms we decided to focus on harnessing the organism to implement biomemristors for more generic use. Of course, as we are a computer music group, we are interested in developing this component with musical applications in mind. However, this does not mean that we are looking for any particular intrinsic relationship between *P. polycephalum*-based processors and music. The biomemristor is intended to be a generic electronic component, in the same way that a programmable digital microprocessor is. Nevertheless, the music processing methods that we have been experimenting with are intended to harness the memristive phenomenon. For

example, in the mapping method we introduced in this paper we explored the fact that the resistance of the component varies in function of voltage that has passed though it in the past. We developed a method in which musical transitions, represented in terms of voltage transitions, are mapped nonlinearly onto music data represented as measurements of current yielded by the memristive behavior of biomemristors.

We would like stress that the music processing scheme introduced above is only of a number of schemes that we have been experimenting with. We are still in the process of investigating the affordance of the biomemristor with respect to how it might be exploited for creating music, possibly in unprecedented ways. What we can say for now is that the biomemristor is essentially a data-driven nonlinear biological processor. It is not clear what advantages might such component bring in the future over currently available computing devices. Still, the musical results we obtained with the biomemristors are comparable with those obtained from previous systems based on sophisticated Artificial Intelligence modeling that have been implemented in our research laboratory, comprising thousands of lines of programming code (McAlpine et al. 1999; Miranda 2004; Gimenes and Miranda 2011); a demonstration is available (Miranda 2016c). This gives convincing evidence that biology can indeed be harnessed to become components of computing architectures, rather than being merely simulated.

The next steps for this research include efforts to gain a better understanding of biological make up of *P. polycephalum*'s with a view on developing effective methods to regulate its memristive properties. Also, we are planning to experiment with networks of biomemristors to establish how their spiking behavior might be harnessed for machine learning; for example, we envisage developing deep learning techniques (Goodfellow *et al.* 2016) with biomemristors.

Is Music Biotech the New Computer Music? In April 2017, Nicholas Negroponte began a talk he delivered at a symposium held at MIT by proclaiming that "Bio is the new digital". He estimated that developments in biotechnology are now at the stage where developments in computing technology were in the middle of the 20th century. Emerging research into synthetic biology, for instance, is increasingly blurring the distinction between artificial machines and biological organisms. The idea of harnessing organisms to develop living machines as if you were programming electronic computers is no longer science fiction. It seems inevitable, therefore, that the field of Computer Music might evolve into something along the lines of Music Biotech. This paper introduced possible pathways for this evolution.

References

Adamatzky, A. (Ed.) 2016. *Advances in Physarum Machines: Sensing and Computing with Slime Mould*. Cham, Switzerland: Springer.

Adamatzky, A. 2010. *Physarum Machines: Computers from Slime Mould*. Singpore: World Scientific.

Adamatzky, A., Costello, B., Melhuish, C. and Ratcliffe, N. 2003. "Experimental Reaction-Diffusion Chemical Processors for Robot Path Planning", *Journal of Intelligent Robotic Systems* 37(3): 233-249.

Braund, E. and Miranda, E. R. (2017). "On Building Practical Biocomputers for Real-World Applications: Receptacles for Culturing Slime Mould Memristors and Component Standardisation", *Journal of Bionic Engineering* 14(2017):151-162.

Braund, E. and Miranda, E.R. (2015). "Music with Unconventional Computing: Towards a step sequencer from plasmodium of Physarum polycephalum". In Johnson C., Carballal A. and

Correia J. (Eds.) *Evolutionary and Biologically Inspired Music, Sound, Art and Design - EvoMUSART 2015*. Lecture Notes in Computer Science, Vol. 9027. Cham, Switzerland: Springer, pp. 15-26.

Caleffi M *et al.* 2015. "On the Solution of the Steiner Tree NP-Hard Problem via Physarum BioNetwork", *IEEE/ACM Transactions on Networking* 23(4):1092-1106.

Choi, I. 1994. "Interactive exploration of a chaotic oscillator for generating musical signals in real-time concert performance", *Journal of the Franklin Institute* 331(6):785-818.

Chua, L.O. 2015. "Everything you wish to know about memristors but are afraid to ask", *Radioengineering* 24(2):319-368.

Chua, L. O. 1971. "Memristor: The missing circuit element", *IEEE Transaction on Circuit Theory* 18(5):507–519.

Doornbusch, P. 2004. "Computer Sound Synthesis in 1951: The Music of CSIRAC", *Computer Music Journal* 28(1):10-25.

Gale, E., Adamatzky, A. and Costelo, B. de L. 2013. "Slime mould memristors", *BioNanoScience* 5(1):1-8.

Gimenes, M. and Miranda, E. R. 2011. "An Ontomemetic Approach to Musical Intelligence". In E. R. Miranda (Ed.), *A-Life for Music: Music and Computing Models of Living Systems*. Middleton, WI: A. R. Editions.

Goodfellow, I., Benjio, Y. and Courville. A. 2016. *Deep Learning*. Cambridge, MA: The MIT Press.

Gotoh, K. and Kuroda, K. (1982). "Motive force of cytoplasmic streaming during plasmodial mitosis of Physarum polycephalum", *Cell Motility* 2(2):173–181.

Gupta, B., Revagade, N. and Hilborn, J. 2007. "Poly (lactic acid) fiber: An overview", *Progress in Polymer Science* 32(4):455–482.

Hiller Jr., L. A. and Isaacson, L. M. 1959. *Experimental Music: Composition with an Electronic Computer.* New York, USA: McGraw-Hill Book Company Inc.

Howard, F. L. 1931. "The Life History of Physarum polycephalum", *American Journal of Botany* 18(2):116-133.

Lerdahl, F. and Jackendoff, R. 1983. "An Overview of Hierarchical Structure in Music", *Music Perception: An Interdisciplinary Journal* 1(2):229-252.

Manning, P. 2013. *Electronic and Computer Music*. Oxford, UK: Oxford University Press.

McAlpine, K., Miranda, E. and Hoggar, S. 1999. "Making Music with Algorithms: A Case-Study System", *Computer Music Journal* 23(2):19-30.

McPherson, A. 2010. "The Magnetic Resonator Piano: Electronic Augmentation of an Acoustic Musical Instrument", *Journal of New Music Research* 39(3):189-202.

Manzo, V. J. 2011. *Max/MSP/Jitter for Music: A Guide to Developing Interactive Music Systems for Education and More*. Oxford, UK: Oxford University Press.

Meyer, R. and Stocking, W. 1970. "Studies on microplasmodia of *Physarum polycephalum* V", *Cell Biology International Reports* 3(4): 321–330.

Miranda, E. R. (Ed.) (2017). *Guide to Unconventional Computing for Music*. Cham, Switzerland: Springer.

Miranda, E. R., Kirke, A., Braund, E. and Antoine, A. (2017). "On Unconventional Computing for Sound and Music". In E. R. Miranda (Ed.), *Guide to Unconventional Computing for Music*. Cham, Switzerland: Springer, pp. 23-62.

Miranda, E. R. and Braund, E. (2017). "A Method for Growing Biomemristors from Slime Mould", *Journal of Visualized Experiments (JoVE).* Available online: https://www.jove.com/video/56076/a-method-for-growing-biomemristors-from-slime-mold (Accessed on 02 November 2017).

Miranda, E. R. (2016a). Biocomputer Rhythms: A composition for piano, percussion and biocomputer by Eduardo R. Miranda. Movie available online: https://vimeo.com/163673832 (Accessed on 02 January 2018).

Miranda, E. R. (2016b). *Music Biocomputing: On harnessing the naturally elegant and efficient problem-solving methods of biological organisms to build musical systems for composition and performance*. Movie available online: https://vimeo.com/163427284 (Accessed on 02 January 2018).

Miranda, E. R. (2016c). Demonstration of the interactive musical computer with pianist Derek Wang. Movie available online: https://vimeo.com/187169524 (Accessed on 04 January 2018).

Miranda, E. R., Admatzky, A. and Jones, J. 2011. "Sound Synthesis with Slime Mould of Physarum Policephalum", *Jornal of Bionic Engineering* 8(2011):107-113.

Miranda, E. R., Bull, L., Gueguen, F., Uroukov, I. S. 2009. "Computer Music Meets Unconventional Computing: Towards Sound Synthesis with In Vitro Neuronal Networks"", *Computer Music Journal* 33(1):9-18.

Miranda, E. R. and Wanderley, M. M. 2006. *New Digital Musical Instruments: Control and Interaction beyond de Keyboard*. Middleton, USA: A-R Editions.

Miranda, E. R. 2004. ""At the Crossroads of Evolutionary Computation and Music: Self-Programming Synthesizers, Swarm Orchestras and the Origins of Melody", Evolutionary Computation 12(2):137-158.

Palotti, P. and Evangelista, G. (2001). "Fractal Additive Synthesis via Harmonic Band Wavelets", *Computer Music Journal* 25(3):22-37.

Parolini, G. 2017. "Music without Musicians … but with Scientists, Technicians and Computer Companies", *Organised Sound* 22(2):286-296.

Pershin, Y. V., La Fontaine, S. and Di Ventra, M. 2009. "Memristive model of amoeba learning", *Physical Review E* 80(2):021926.

Risset, J.-C. 1985. "Computer Music Experiments 1964 - …", *Computer Music Journal* 9(1):11-18.

Roads, C. 1980. "Interview with Max Mathews", Computer Music Journal 4(4):15-22.

Schaeffer, P. 1971. "Music and Computers", *Proceedings of UNESCO's Music and Technology Meetiong in Stockholm*. Paris, France: La Revue Musicale.

Strukov, D. B., Snider, G. S., Stewart, D. R. and Williams, R. S. 2008. "The missing memristor found", *Nature* 453(7191):80-83.

Trefzer, M. A. 2017. "Memristor in a Nutshell". In E. R. Miranda (Ed.), *Guide to Unconventional Computing for Music*. Cham, Switzerland: Springer, pp. 159-180.

Versace, M. and Chandler, B. 2010. "The brain of a new machine", *IEEE spectrum*, 47 (12): 30–37.

Volkov, A. G., Reedus, J., Mitchell, C. M., Tucket, C., Forde-Tuckett, V., Volkova, M. I., Markin, V. S. and Chua, L. (2014). "Memristors in the electrical network of Aloe vera L", *Plant Signaling & Behavior* 9(7):e29056.

Weinberg, G. and Driscoll, S. (2006). "Toward Robotic Musicianship", *Computer Music Journal* 30:4(28-45)

Xenakis, I. 1972. *Formalized Music*. Boomington, IN: Indiana University Press.

Appendix

Event	Input	n	V	ϕ	$I \times 10^{-4}$	ΔI	Output
1	71	1	0.00	0.00	+0.0252	0.00	80
2	80	1	5.00	+5.00	+0.1961	678.17	69
3	80	2	3.33	+8.33	+0.2053	4.69	81
4	78	1	7.50	+0.83	-0.0286	113.93	66
5	76	1	8.00	-7.17	-0.3147	1000.34	71
6	78	2	6.66	-13.83	-0.5196	65.10	80
7	81	1	8.57	-5.26	-0.1758	66.16	78
8	81	2	7.50	-12.76	-0.2810	59.84	78
9	83	1	8.88	-3.88	-0.0967	65.58	80
10	80	3	7.00	-10.88	-0.3313	242. 60	73
11	81	3	7.27	-3.61	-0.1187	64.17	78
12	78	3	7.50	+3.89	+0.1312	210.53	75
13	71	2	8.46	-4.57	-0.1433	209.23	68
14	78	4	7.14	-11.71	-0.4152	189.74	66
15	80	4	7.33	-4.38	-0.0965	76.75	78
16	81	4	7.50	+3.12	+0.1725	278.75	83
17	78	5	7.05	+10.17	+0.3013	74.66	80
18	80	5	7.22	+2.95	+0.0578	80.81	78
19	76	2	8.94	-5.99	-0.2711	569.03	71
20	78	6	7.00	-12.99	-0.4634	70.93	80
21	75	1	9.52	-3.47	-0.0489	89.44	76
22	76	3	8.63	-12.10	-0.3168	547.85	71
23	71	3	8.69	-3.41	-0.0491	84.50	68
24	68	1	9.58	+6.17	+0.2969	704.68	64
25	64	1	9.60	-3.43	-0.1660	155.91	68

26	68	2	9.23	-12.66	-0.4230	154.81	71
27	71	4	8.51	-21.17	-0.7962	88.22	78
28	76	4	8.57	-12.60	-0.5334	33.00	78
29	80	6	7.93	-20.53	-0.6975	30.76	81
30	69	1	9.66	-10.87	-0.3360	51.82	80
31	80	7	7.74	-18.61	-0.6643	97.70	69
32	78	7	7.81	-10.80	-0.3274	50.71	80
33	76	5	8.48	-2.32	-0.0662	79.78	71
34	78	8	7.64	-9.96	-0.3064	362.83	75
35	80	8	7.71	-2.25	-0.0581	81.03	78
36	81	5	8.61	+6.36	+0.2386	510.67	83
37	78	9	7.56	+13.92	+0.3568	49.53	80
38	81	6	8.42	+5.50	+0.1604	55.04	78
39	71	5	8.71	-3.21	-0.1527	195.19	76
40	80	9	7.75	-10.96	-0.3456	126.32	73
41	73	1	9.75	-1.21	+0.0217	106.27	78
42	78	10	7.61	-8.82	-0.2899	1435.94	66
43	66	1	9.76	+0.94	+0.1159	139.97	75
44	75	2	9.54	+10.48	+0.1223	5.52	76
45	76	6	8.66	+19.14	+0.5357	338.02	71

Table A1. Data produced by the pitches of J. S. Bach's 'Gavotte en rondeau', with step dwell time of 2s and 50% offset percentage.