Oscillation-Based Slime Mould Electronic Circuit Model for Maze-Solving Computations

Vasileios Ntinas, Ioannis Vourkas, *Member, IEEE*, Georgios Ch. Sirakoulis, *Member, IEEE*, and Andrew I. Adamatzky

***Abstract*—The ability of slime mould to learn and adapt to periodic changes in its environment inspired scientists to develop behavioral memristor-based circuit models of its memory organization. The computing abilities of slime mould Physarum polycephalum have been used in several applications, including to solve mazes. This work presents a circuit-level bio-inspired maze-solving approach via an electronic model of the oscillatory internal motion mechanism of slime mould, which emulates the local signal propagation and the expansion of its vascular network. Our implementation takes into account the inherent noise existent in the equivalent biological circuit, so that its behavior becomes closer to the non-deterministic behavior of the real organism. The efficiency and generality of the proposed electronic computing medium was validated through SPICE-level circuit simulations and compared with data from two cardinally different biological experiments, concerning 1) enhancing of Physarum's protoplasmic tubes along shortest path and 2) chemo-tactic growth by diffusing chemo-attractants.**

***Index Terms*—Labyrinth, memristor, oscillation, slime mould, SPICE.**

I. Introduction

THE primitive intelligence of Physarum polycephalum (slime mould) is mostly demonstrated during its vegetative stage when it turns into plasmodium, a yellowish vascular network which expands in search of nutrition [1]. The protoplasm moves in an oscillatory manner with emphasis on the direction of the nutrition source from which it receives the strongest chemical stimulus. Most importantly, in the presence of many nutrition sources the vascular network of the plasmodium is expanded to reach all of them simultaneously and then it shrinks until it achieves the shortest possible interconnections between the sources [2]. Nakagaki *et al.* [3] were the first to show experimentally that this particular property of Physarum could be used in unconventional computing approaches to maze-solving applications. Later, many more biological experiments followed, exploiting the behavior of Physarum in several problems including path planning [4], maze-solving [5], game theory [6], deadlock avoidance [7], traveling salesman [8], etc.

The internal oscillations of Physarum's electrical potential and cytoplasm shuffling is one of the possible routes towards achieving synchronisation of its distant parts and distributed decision making. Laboratory measurements presented in [9] demonstrated hysteretic current-voltage characteristics of the protoplasm part of Physarum which are consistent with ideal (or active) memristors. Moreover, according to experimental results [10], the slime mould learns periodic changes in the environment, memorizes their periods, and adjusts its behaviour accordingly. Inspired by such observations, Traversa *et al.* [11] presented adaptive circuits comprising LC contours with damping resistors and memristors [12], [13], later tested experimentally in [14], to model memory organization and predict possible biological responses by simple organisms with internal memory abilities. Circuit models constitute low-cost and fast prototyping alternative solutions to expensive and time-consuming biological experiments and, in this direction, [11] presented an early approach of memristive adaptive electronic circuits targeting bio-inspired information processing applications. Accurate circuit models are definitely expected to play an important role in the study of behavioral mechanisms and the exploitation of the computing capabilities of Physarum in hardware.

As a proof of concept of its computing abilities slime mould *Physarum polycephalum* was used to solve mazes. The maze problem solution, especially its physical realization, attracted attention of engineers and scientists for over 60 years. First published results in this field Shannon's mechanical mouse maze solver [15]. Shannon's mouse did not calculate a shortest path in the maze, it was rather labelling already visited sites and continued exploring maze till a central chamber was found. Maze solving is associated with a shortest path problem, therefore we consider prototypes designed for either of them. Further experimental laboratory prototypes of maze solvers were based on physical properties of a "computing substrate" filling the maze. Wave propagation in excitable chemical medium—the Belousov–Zhabotinsky reaction—was exploited in few versions of the chemical computers for a shortest path problem [16]–[20]. Given an excitation wave initiated at point x the wave will reach a point y along a shortest path, navigating around obstacles. To compute the shortest path we must record profiles of excitation waves propagating from x to y. Then put medium to the rest, excited site y and record profiles of waves till they reach x. Intersection of profiles of waves traveling from x and y with profiles of waves traveling from y to x gives us the shortest path. Propagation of physical disturbances of a

Manuscript received December 13, 2015; revised March 22, 2016; accepted April 20, 2016. This work was supported in part by CONICYT FONDECYT POSTDOCTORADO No. 3160042/2016 Chilean government research grant. This paper was recommended by Associate Editor T. S. Gotarredona.

V. Ntinas and G. Ch. Sirakoulis are with the Department of Electrical and Computer Engineering, Democritus University of Thrace, Xanthi 67100, Greece (e-mail: vntinas@ee.duth.gr; gsirak@ee.duth.gr).

I. Vourkas is with the Centro de Investigación en Nanotecnologia y Materiales Avanzados, Department of Electrical Engineering, Pontificia Universidad Católica de Chile, Santiago P.C., 7820436, Chile (e-mail: iovourkas@uc.cl).

A. I. Adamatzky is with the Unconventional Computing Centre, University of the West of England, Bristol, BS16 1QY, U.K. (e-mail: andrew.adamatzky@uwe.ac.uk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSI.2016.2566278

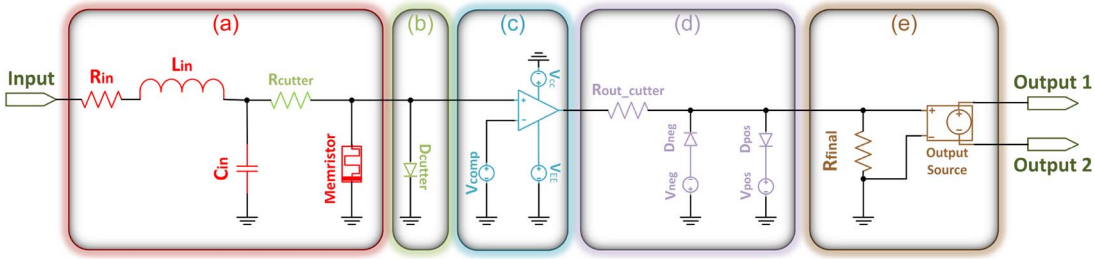


Fig. 1. Slime mould electronic circuit model. (a)–(e) highlight the five different parts of the circuit, whose role is explained in the text.

computing medium is used in other prototypes of maze solver. Thus, electrical current propagates along the path with minimal total resistance. The intensity of the current can be visualized via intensity of illumination in a gas discharge system [21], [22]. A shortest path approximated by gradients of chemical diffusion from the destination site is visualized by chemo-sensitive droplets [23], [24], worms [25] or slime mould [5]. A shortest path approximated by a fluid flow from hot source to cold destination is visualized by dye particles, whose intensity is proportional to speed of the flow [26]. Path computation is done by propagating disturbances and visualization is done by external agents. Only two prototypes store configuration of shortest paths for a long time: gas-discharge maze solver [21], [22] and slime mould maze solver [3], [5], other than show the path just momentarily.

In regards to the slime mould *Physarum polycephalum* computational abilities, it can store results of computation for longest time (days) with minimal supply of energy. First experimental slime mould maze solver [3] was inspired by and heavily relied on original Belousov–Zhabotinsky maze solver by Steinbock *et al.* [16]. The slime mould was placed in all sites of the maze. Sources of nutrients were placed in two sites to be connected by a path. Due to the increasing flow of cytoplasm between parts of slime mould which occupy the sources of nutrients, the protoplasmic tubes lying along the shortest path between them increase in diameter while the tubes along non-shorter paths decrease in diameter or are abandoned. Thus, a shortest path is represented by the thickest protoplasmic tube. A closest physical analogy could be a river bed formation, if we would be allowed to change direction of water streams. The slime mould maze solver implemented in [5] exploits diffusion of attractants and chemo-tactic behavior of slime mould. A source of attractants is placed in the central chamber of the maze. The attractants diffuse in agar gel in the channels of the maze. A piece of Physarum is placed in some other site of the maze. The Physarum grows predominantly towards the source of attractants. When the slime mould reaches the source of attractants we have the start point and the destination point connected by a thick protoplasmic tube, which represents a path between the points. Not just sources of nutrients can be used in the setup [5] but any strong chemo-attractants [27].

In the present study we build upon our previous work in [28], which presented a thorough exploration of the circuit properties and learning sensitivity of memristive adaptive filters [11], [28]. Here we use a memristive LC contour in an enhanced electronic circuit model of the internal oscillatory motion mechanism of the plasmodium, aiming to emulate the local signal propagation and the expansion of its vascular network during the reproduction stage within complex maze environments. To this end, we create a network of such circuit modules, thereby extending the study/usage of the proposed circuit as filter in [28], which we then interconnect properly according to specific maze patterns. In such a way, the proposed here oscillation-based slime mould electronic circuit is different from the other memristor based maze solutions introduced earlier in literature, such as [29] and [30], where the application of memristive devices in maze solving problems has been also implemented by grids of memristors but without any development of any biologically inspired oscillation features. The proposed circuit is then operated in the region of maximum adaptation (learning), so every module adjusts its function depending on the characteristics of the input waveform and network expansion corresponds to gradual training of subsequent interconnected modules. We prove the computing capabilities of the proposed circuit model using two cardinally different experimental case studies: enhancing of Physarum's protoplasmic tubes along shortest path [3], where the thickness of the Plasmodium tube is proportional to flow, i.e., feedback; and chemo-tactic growth of slime mould, where Physarum acts as a concurrent navigator (several growing zones) by diffusing chemo-attractants [5]. In addition, the proposed implementation introduces inherent noise abilities to the equivalent biological circuit that can result, beside the optimal, to several maze solutions which is closer to the non-deterministic behavior of real organisms shown in the corresponding biological experiments [3], [5]. A simulation-based verification of the proposed computing circuit is provided using the Cadence SPICE environment, based on a SPICE-compatible device model of a bipolar voltage-controlled threshold-type memristor [31]. Our work constitutes an early attempt to implement an accurate enough electronic laboratory prototype of Physarum-based computing machines [1].

II. Electronic Circuit Model of Slime Mould Adaptation and Movement

Inspired by our own observations of periodicity in respective slime mould biological experiments [32], we designed an electronic circuit which models the adaptation and movement of the slime mould according to the changing environmental conditions. In Fig. 1 we define the basic parts/stages of this main circuit module, which comprises a memristive LC contour combined with additional electronic circuit components. With

this circuit unit we represent a small quantum of the slime mould protoplasmic tube, in order to model local signal propagation through oscillations within a complex network of interconnected circuit units, which generally models the network development of the biological organism during its reproduction. At this stage, the slime mould has the form of a plasmodium which is able to sense the existence of nutrient sources and head towards them, while increasing the intensity of internal protoplasmic oscillations in the desired direction. The specific stages of the main circuit unit, shown in Fig. 1, are as follows:

a) LC filter with memristive damping.
b) Positive voltage clipper across the memristive device.
c) Comparator for signal conversion.
d) Square voltage pulse clipper.
e) Output.

Stage #a reflects the internal memory abilities and the biological oscillatory system of the slime mould, based on which it recognizes periodic changes in the environment, memorizes their periods, and adjusts its future behavior according to the stored information [1], [10]. It consists in a simple bandpass filter, where the capacitor and the memristive device are connected in parallel so that the output response of the filter will depend on the time-history of the voltage and frequency of the applied input signal [11], [14], [28], [33]. To understand the operation of this part, we should first focus on the behavior of its memristive component. The main idea is to use the changing memristance in order to exploit the oscillations in the LC contour.

We assume a voltage-controlled, threshold-type switching bipolar memristive device, whose behavior is generally described by the following nonlinear constitutive relation:

$$I_M(t) = G(x) \cdot V_M(t). \tag{1}$$

The used model [31] attributes the resistance-switching effect to the modulation of an effective tunneling-distance x (state-variable). G is the conductance of the device, whereas I_M and V_M represent the flowing current and the applied voltage, respectively. The resistance-switching rate is very slow below (or very fast above) the voltage thresholds $V_{\rm SET}$ and $V_{\rm RESET}$, which here are considered as asymmetric, i.e., $|V_{\rm SET}| \neq |V_{\rm RESET}|$. Inspired from the original circuit model proposed by Hewlett–Packard Labs for TiO2-based devices in [34], this model concentrates on the tunneling-resistance of the undoped dioxide layer and approximates the overall memristance M of the device, as follows:

$$M(x) = \frac{1}{G(x)} = f_0 \cdot \frac{e^{2x}}{x}. \tag{2}$$

Equation (2) contains a model-fitting constant f_0 and gives the memristance for a certain value-range of the state variable x. A heuristic equation for the expected response of x as function of V_M is given as follows:

$$x = x_0 \cdot \left(1 - \frac{m}{r}\right). \tag{3}$$

Parameter x_0 is the maximum value of x. The term in parenthesis of (3) contains a voltage-dependent parameter r and a fitting constant m, which both affect the boundaries of the barrier width. Parameter r also defines the current state of the device. Its value is monitored and maintained within a valid range $r_{\rm MIN} \leq r \leq r_{\rm MAX}$. Consequently, the memristance M is correspondingly maintained between two boundary values $R_{\rm ON} \leq M \leq R_{\rm OFF}$ according to (2). The modulation-rate of x depends on the applied voltage. Such assumption is encapsulated in the time derivative of r, which is described as follows:

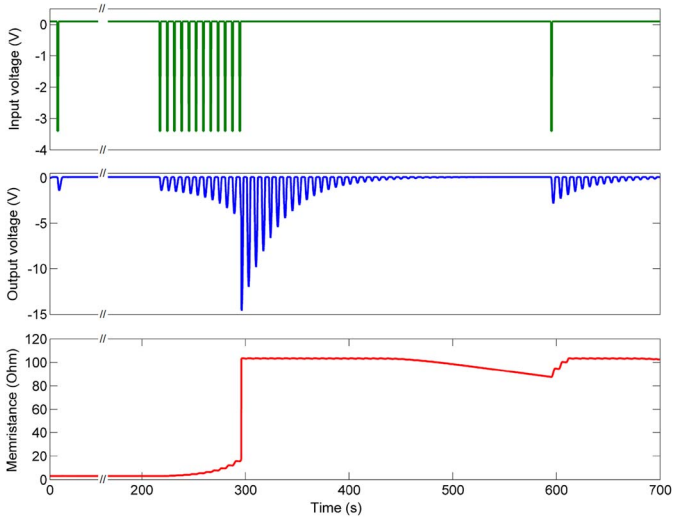


Fig. 2. Simulation of the circuit of stage #a trained by a periodic sequence of pulses with separation $T \approx 7$ s. In simulations we set $R, L, C = 0.1\ \Omega$, 4 H, 0.3 F, $\{R_{\rm ON}, R_{\rm OFF}\} = 3, 103\ \Omega$, $V_{\rm RESET} = 4.5$ V, and $\{V_p, V_o\} = \{0.1, 3.4\}$ V.

$$\dot{r} = \begin{cases} a_{\rm RESET} \cdot \frac{V_M + V_{\rm RESET}}{c + |V_M + V_{\rm RESET}|}, & V_M < V_{\rm RESET} \\ b \cdot V_M, & V_{\rm RESET} \leq V_M \leq V_{\rm SET} \\ a_{\rm SET} \cdot \frac{V_M - V_{\rm SET}}{c + |V_M - V_{\rm SET}|}, & V_M > V_{\rm SET}. \end{cases} \tag{4}$$

Parameters $a_{\rm RESET}$, $a_{\rm SET}$, b, and c of (4) are fitting constants that shape the intensity of the state variable dynamics. Herein we will assume that a positive/negative voltage applied to the top terminal (w.r.t. the bottom terminal defined by the thick line in the schematic) of a memristive device will decrease/increase its memristance.

This relatively simple circuit of stage #a exhibits learning capabilities in response to a train of voltage pulses of selected frequencies. Specifically, a small positive applied voltage V_p tends to switch the memristive device ON, so that the LC contour becomes damped and the excited oscillations decay fast (untrained state). However, as shown in the simulation results of Fig. 2, when this circuit is subjected to a series of negative pulses of amplitude V_o, which are on resonance frequency $1/(LC)^{1/2}$ (training state), the amplitude of voltage oscillations on the capacitor increases with each pulse and at some point exceeds $V_{\rm RESET}$ of the memristive device, which in turn switches to $R_{\rm OFF}$. In our previous work [28] we studied the properties of this circuit and experimented with its sensitivity to identify the regions of maximum learning and to enable memorization of multiple training frequencies. Indeed, the circuit is more sensitive to frequencies belonging to a narrow fixed range, whereas off-resonance signals have less effect on the memristor(s). For example, according to [32], the resonant

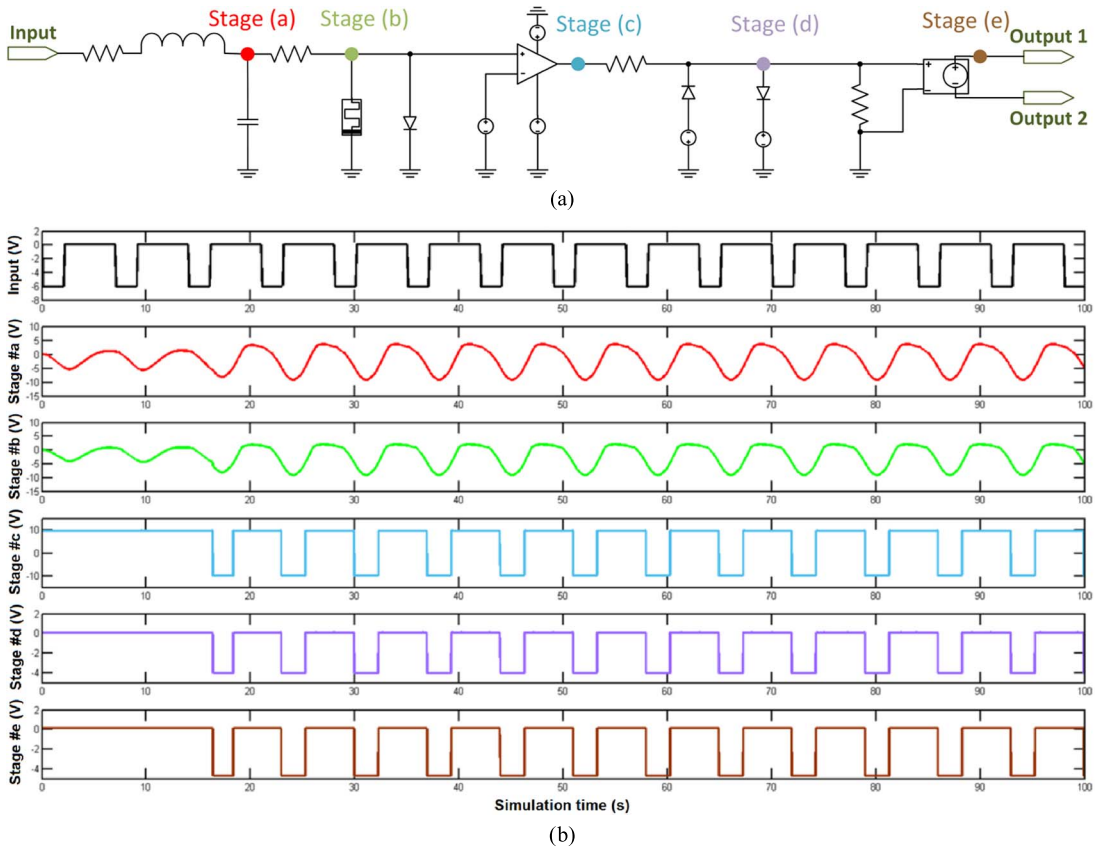


Fig. 3. SPICE simulation results of the proposed circuit module. (a) Shows the output point of every stage of the circuit where the voltage was measured during simulation and is then demonstrated in the corresponding graph of (b), using $V_o = -6$ V, $V_P = 0.1$ V, T = 7 s and input pulse duration 2 s.

period of slime mould response was found to be around 70 s which, in our circuit simulations, we decreased to only 7 s to shorten the model verification process. Moreover, the circuit response to periodic input signals also depends on their amplitude V_o. Smaller amplitudes do not cause significant switching since the voltage drop across the memristive element does not exceed $V_{\rm RESET}$ [28], [31]. Consequently, with the LC contour being less damped, the output oscillations caused by a single input pulse survive longer (trained state) and then, practically, the output voltage can be passed to the input of a subsequent connected circuit module. In this direction, stages #b through #e modify the output of stage #a so that it becomes suitable to be applied to the input of a subsequent module. Moreover, they isolate and secure the proper and stable individual function of every quantum of the entire slime mould network.

More specifically, the positive voltage clipper in stage #b prevents any unwanted state-change from $R_{\rm OFF}$ to $R_{\rm ON}$ by restricting the positive voltage drop on the memristor. For correct functioning, we also include a series resistor $R_{\rm cutter} = 1\ \Omega$ before the memristor, and with the selected values for $R_{\rm IN}$, L, C, M we achieve positive voltage clipping at approximately 2 V. Right afterwards, stages #c and #d convert the voltage on the memristors to a form appropriate to be used as input for the subsequent circuit module. In fact, stage #c includes an operational amplifier (opamp) configured as a comparator, which converts the oscillating input signal to a train of square pulses of known maximum amplitude, based on the power supply (here $V_{CC} = 10$ V and $V_{EE} = -10$ V). Particularly, negative square pulses are generated when the positive input of the opamp is lower than the negative one. So, a voltage source $V_{\rm comp} = 4.5$ V is used to permit the generation of negative square pulses only after the circuit in stage #a has been trained. Next, a voltage clipper is used in stage #d to limit the amplitude of the created square voltage pulses within proper bounds (here 0.3 V and -4.5 V) using $V_{\rm neg} = 3.3$ V, $V_{\rm pos} = -0.6$ V, and $R_{\rm out\_cutter} = 1$ KΩ. Finally, stage #e consists of a load resistor $R_{\rm final} = 10$ KΩ and a voltage-controlled voltage source. The latter provides the necessary isolation between the output and the input of connected circuit modules, removing the effect that the L and C elements of a particular module have over the output of the previous connected module. The output voltage source is activated only after the circuit in stage #a has been trained. The two available output terminals facilitate the series connection of output voltage sources of different interconnected modules, something necessary when modeling branching points of protoplasmic tubes.

III. PSPICE Implementation

A. Main Module Description

Since threshold-type switching is closer to the actual behavior of most experimentally realizable memristive devices, throughout this work we base our circuit simulations on the SPICE-compatible, threshold-type device model of a voltage-controlled bipolar memristor, which we summarized in the

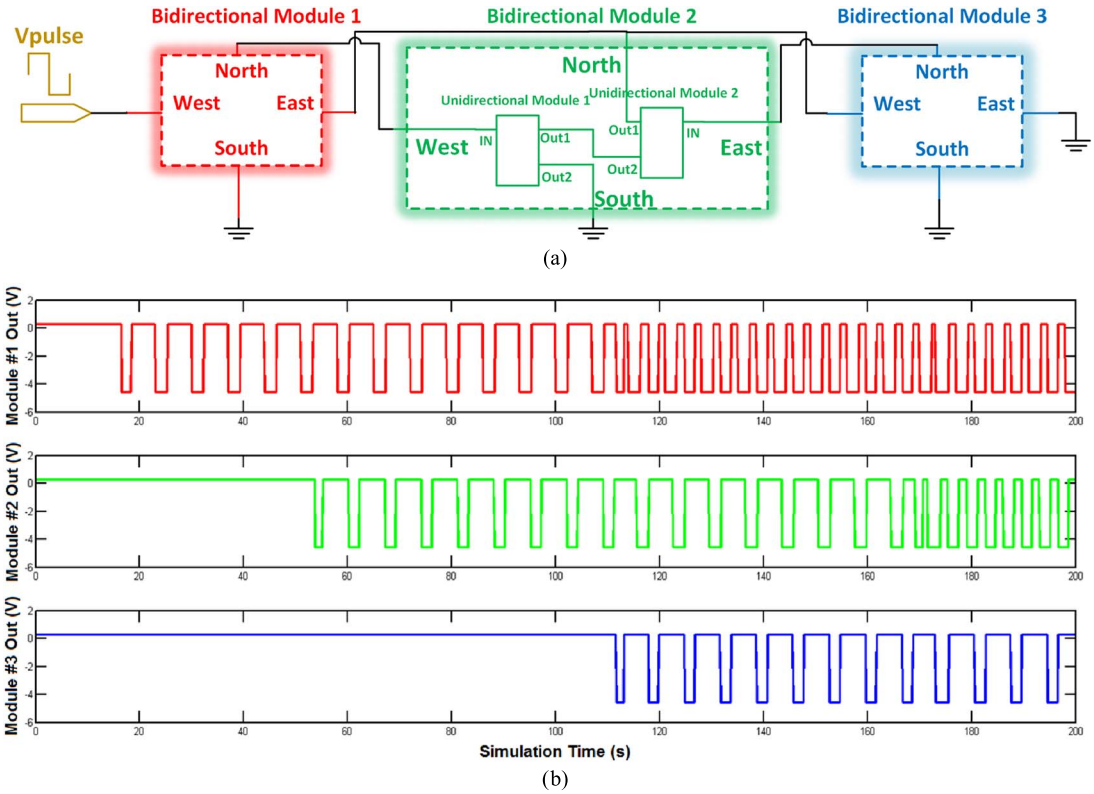


Fig. 4. Bidirectional signal propagation circuit model. (a) A small network interconnection example comprising three composite circuit modules, emphasizing both the external E, W, S, N and the internal interconnection of the unidirectional modules. (b) SPICE simulation results demonstrating the cumulative output voltage at terminal "N" of every module.

previous section [31]. The values of the parameters of the model are set as given in [31] with $\{R_{\rm ON}, R_{\rm OFF}\} = \{3, 250\}$ Ω and $\{V_{\rm SET}, V_{\rm RESET}\} = \{10, -4.5\}$ V.

Using the aforementioned model for memristors, the slime mould behavioral circuit model of Fig. 1 was implemented in the Cadence PSPICE environment as a compact circuit module to facilitate the formulation of complex network configurations which represent the target problem-space. Apart from the memristor device model, the rest of circuit components are found in the standard libraries of the circuit simulation environment. Additionally, a resistor $R_{\rm Diode}$ of very large value was connected in parallel to every diode for better performance. The complete SPICE netlist of the slime mould sub-circuit, named as "RLCM," is given in Table I in the Appendix, whereas Fig. 3 shows the simulation results of the circuit module using $V_o = -6$ V, $V_P = 0.1$ V, T = 7 s and input pulse duration 2 s, demonstrating particularly the voltage measured at the output of every different stage of the circuit.

B. Composite Model for Bidirectional Signal Propagation

Several experiments have shown that, as part of its foraging strategy to discover and exploit new resources, an amoeboid organism grows in the form of an interconnected protoplasmic tubular network where the grow speed depends on the environmental conditions; in our case, the externally applied voltage [10], [32]. It initially expands forming a relatively coherent foraging front so as to maximize the total searched area. Finally, the protoplasmic network is optimized to provide a robust and speedy transportation of nutrients and metabolites in the plasmodium body [2]–[5].

In order to comply with the spreading mechanisms of slime mould, we use pairs of circuit modules which are connected in series via their output terminals, as shown in Fig. 4(a), thus modeling bidirectional signal propagation within the emulated protoplasmic veins. The composite circuit module (combining two main units) comprises four terminals in total: two input terminals (east and west), a single cumulative voltage output terminal (north), being equivalent to two controlled voltage sources connected in series, and a fourth terminal (south) which serves as a minimum voltage reference and is normally grounded. Each one of the internal unidirectional circuit modules is activated independently whenever it has been trained by the signal applied to its input.

More specifically, in Fig. 4(a) the cumulative output of the module #1 (module #3) is driven to the left (right) input of module #2, whereas the cumulative output of module #2 is in turn communicated to the right (left) input of module #1 (module #3). In this specific circuit snapshot, where the input voltage is applied to the left input of module #1, the signal will propagate gradually from left to right through modules #2 and #3, respectively. Similarly, if the input signal had been applied to any of the inputs of the central module (module #2), it would have been communicated simultaneously to the corresponding inputs of modules #1 and #3, thus spreading in both possible directions. The simulation results in Fig. 4(b) confirm the sequence in which the three interconnected modules are trained. We also note that the output voltage pulse train of

modules #1 and #2 is modified after a short period of time, which is attributed to whether only one or both of the internal unidirectional circuit modules are trained. When both of them are trained, due to possible phase shift based on the time when the second circuit module gets trained and activates its output, the summation of the two output signals may lead to such different cumulative voltage pulses or to pulses with twofold amplitude. Nevertheless, this does not affect the overall and desired operation of the computing medium. Due to the nature of the target application, it is not necessary to form circuit modules who accept more than two inputs (i.e., have more than two neighbors), except when modeling route intersections, which are treated as special cases and are discussed in detail later.

C. Application Mapping and Computing Methodology

Using the previously presented circuit modules, the shortest path computing process consists in the following steps:

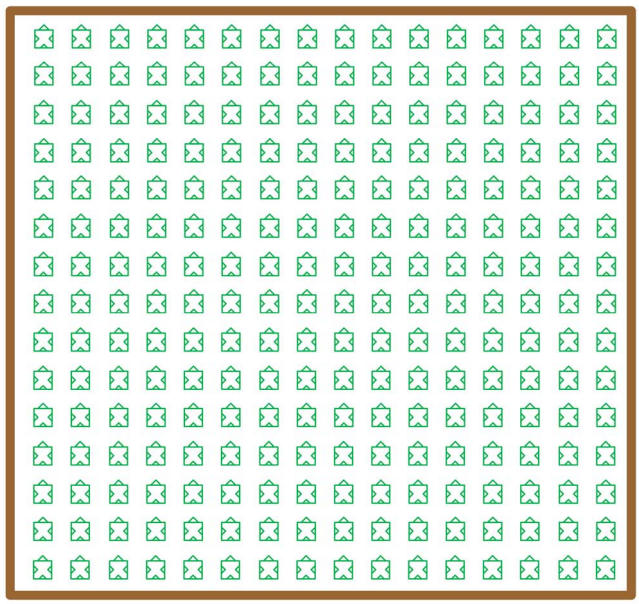
1) Mapping of the target-problem space (maze) on a 2-D circuit network by proper interconnection of the circuit modules.
2) Definition of start and end point(s).
3) Application of input signal to the start point(s).
4) Execution until the circuit module at the end point gets trained.
5) Study of the output time-sequence of the modules at every intersection met while searching backwards, from the end point(s) towards the start point(s).
6) Definition of the shortest path(s).

Computation ends when the RLCM part of the destination module(s) is trained. The execution time should be long enough for the signal propagation to reach the destination (exit) point of the labyrinth through any possible path(s). Reading the results of computation consists in tracing the signal propagation backwards, starting from the exit point. The main interest is in the intersections of the computing medium. In fact, in such points we are able to figure out the direction of the signal which first reached the intersection module by monitoring the time sequence of the output signals of the neighboring modules in all possible directions. Of course, if two signals reach the intersection module simultaneously, it means that there are two similar paths to follow. Therefore, by repeating this strategy and following the shortest path after every intersection, we soon reach the entrance module and hence discover the optimal solution to the problem. In the following sections we evaluate the efficiency of the proposed HW-based computing methodology using data from two different Physarum maze-solving experimental case studies.

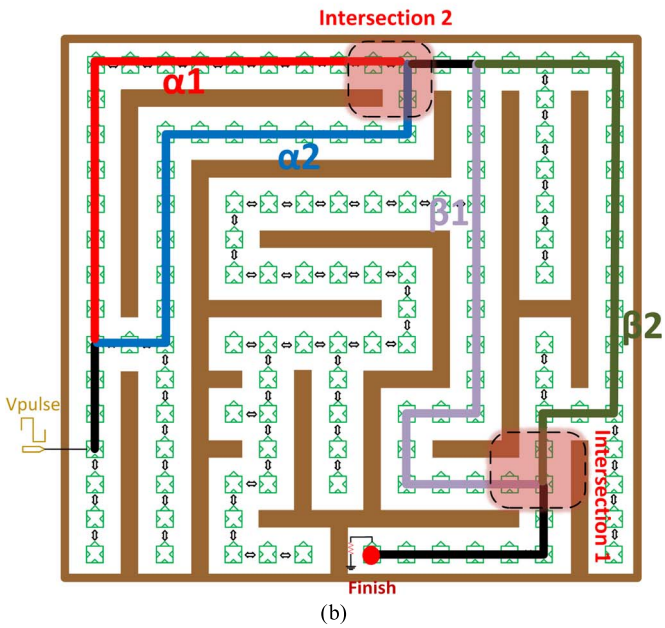
IV. Solution of a Square Labyrinth

A. Biological Experiment

In [3] Nakagaki *et al.* proved experimentally that the slime mould's optimization of its protoplasmic tube network can be interpreted as a computation. In their experiments the slime mould visualizes the minimum-length solution between two points in a square labyrinth, in which they previously placed two nutrient blocks, with its thickest protoplasmic tube. The



(a)



(b)

Fig. 5. (a) A 15 × 16 2-D grid of circuit modules. (b) Maze plan mapped on the 2-D circuit grid. $\alpha 1$, $\alpha 2$, $\beta 1$, and $\beta 2$ denote the different paths between the highlighted intersections of interest.

slime mould, initially filling the entire surface of the maze, four hours later formed a single thick tube covering, in the majority of the repeated experiments, the shortest distance between the two food sources. The nature of this experiment is different from the proposed computing methodology, which emulates the expansion of the plasmodium from the start point following the chemical stimulus of the nutrient source at the end point. Nevertheless, it is a classical experiment, the first of its kind, so we use it to show biological background of our computing process.

B. Electronic SPICE Modeling and Simulation

Using the slime mould electronic module, we developed an electronic computing model to emulate the biological experiment with the rectangular labyrinth [3]. We first mapped the

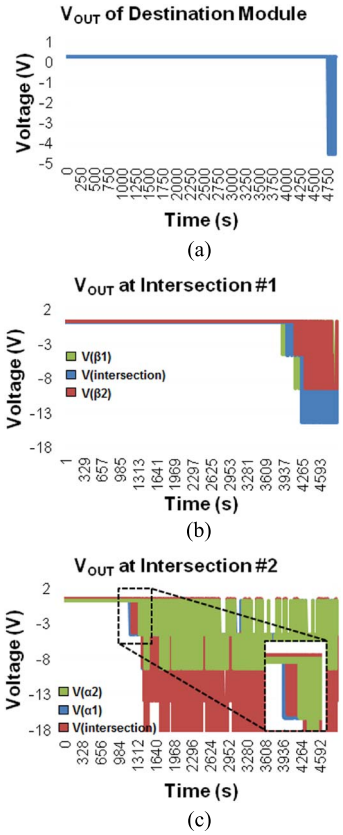


Fig. 6. SPICE $V_{\mathrm{OUT}} - t$ simulation results concerning (a) the exit point's output voltage, (b) the output voltage of the modules from routes $\beta 1$ and $\beta 2$ connected to the central module of the first intersection, and (c) the output voltage of the modules from routes $\alpha 1$ and $\alpha 2$ connected to the central module of second intersection.

labyrinth on a 15×16 square grid of slime mould circuit modules, matching exactly the floor plan of the conducted experiment, as shown in Fig. 5(a). Wherever walls are found, we left the modules unconnected (for readability reasons these areas in Fig. 5(b) are covered with brown color). The rest of the modules were properly interconnected, according to the rules described previously. However, unlike in simple path modeling, as shown in Fig. 5, in every intersection modeling we also need to use the south terminal of the intersection module as an additional input, so it is not grounded as in Fig. 4. This allows summing the output signals of more than two paths, arriving at the intersection, and their propagation to the next module(s). Nevertheless, it is important to note that any signal applied to the south terminal of a bidirectional intersection module is directly connected (added) to its cumulative output. Therefore, this way the output of a bidirectional module is also indirectly connected to its own input via the south terminal of the intersection module. Consequently, to avoid this kind of unwanted connections we include an additional unidirectional module between the south terminal of the intersection module and the output of the connected bidirectional module(s).

In order to emulate the placement of sources of nutrients in the biological counterpart, we apply a sequence of 2-s-wide $\{V_o, V_p, T\} = \{-8\ \mathrm{V}, 0.1\ \mathrm{V}, 7\ \mathrm{s}\}$ negative near-resonant square shaped voltage pulses to the input of the slime mould

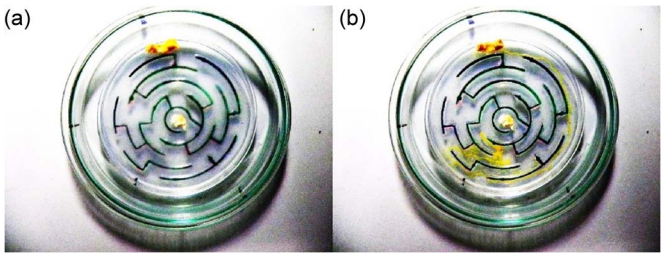


Fig. 7. (a) Starting state of experiment with Physarum at the exterior level. (b) The largest coverage of the Physarum searching path towards the center of the labyrinth.

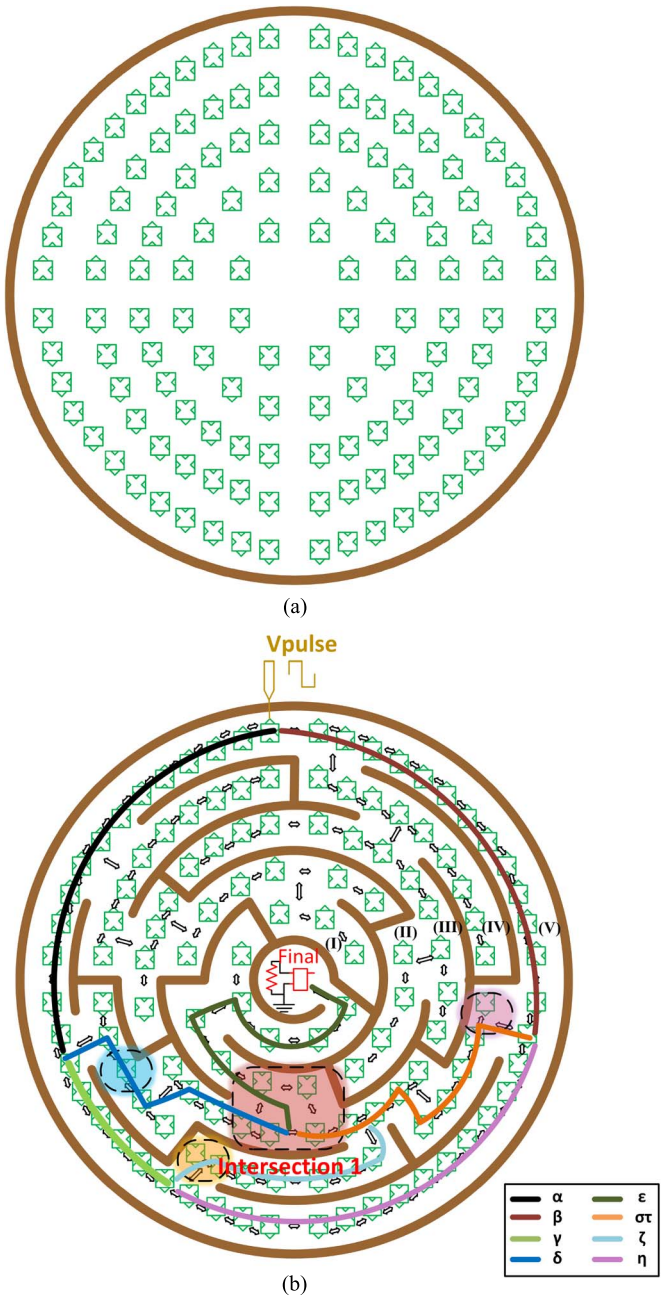


Fig. 8. (a) A 2-D circular grid of circuit modules. (b) Maze plan mapped on the 2-D circuit grid. Greek letters α through η and colors in the legend denote the different path sections between the highlighted intersections of interest. The shortest path given by our simulations consists of the sections α (black), δ (blue), and ϵ (dark green).

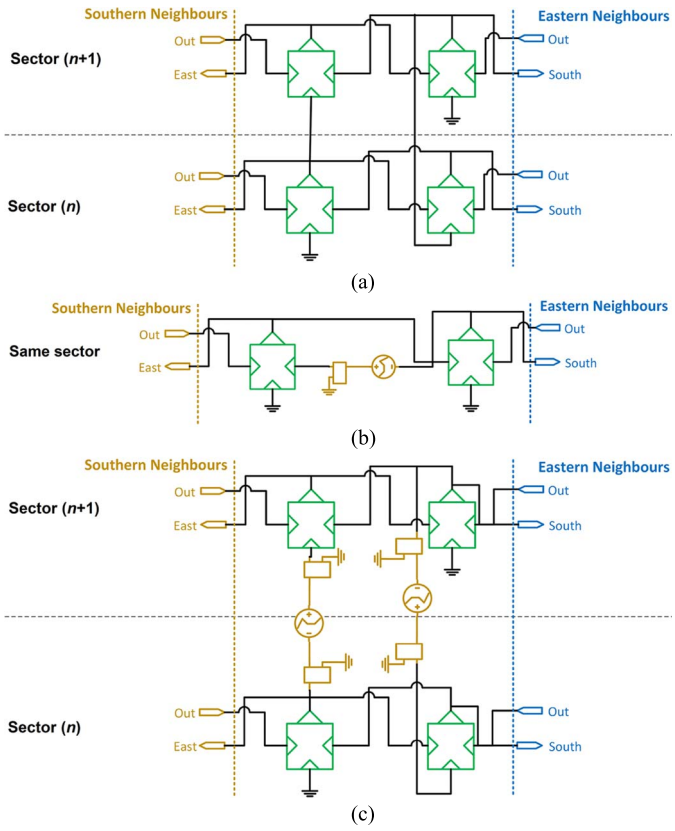


Fig. 9. (a) Intersection wiring example between bidirectional modules of two consecutive sectors without noise generator. (b) Typical interconnection between bidirectional modules of the same sector with noise generator. (c) Intersection circuit modeling between bidirectional modules of two consecutive sectors with noise generator.

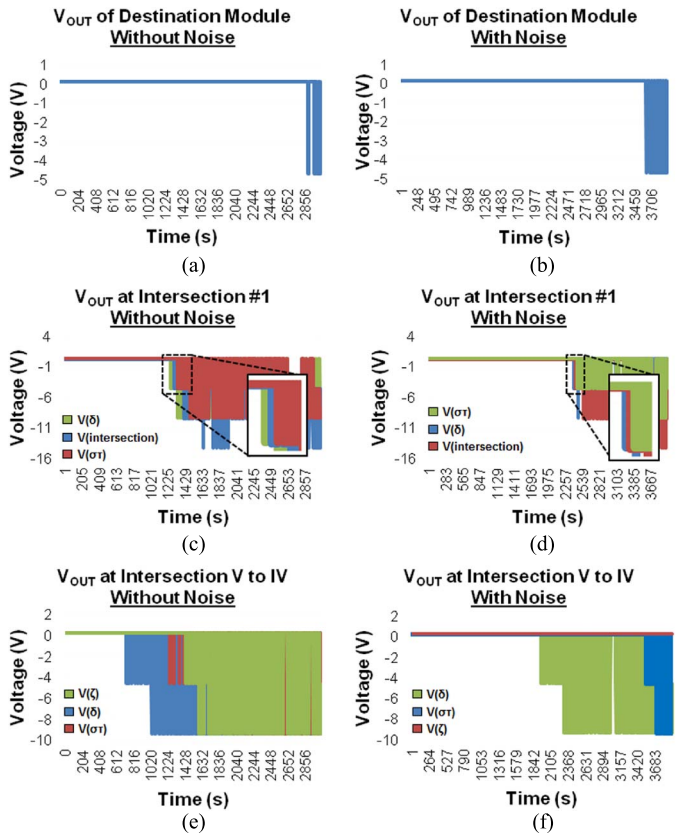


Fig. 10. $V - t$ graphs concerning the cumulative output voltage of the circuit module at (a, b) the exit point, (c, d) at the first intersection, and (e, f) at the intersection passing from sector V to IV, concerning (a, c, e) no noise sources or (b, d, f) after noise sources have been inserted.

electronic module at the entrance (start point) of the labyrinth. Unlike in the biological experiment, where there is another nutrient source at the exit point, here we avoid using an additional voltage source because leaving the exit module floating permits monitoring the corresponding memristor state and thus realizing when computation is completed. With an extra voltage source at the exit point the total computation time would be half the initial one, but would pose problems in the reading strategy of the computation result.

In the conducted SPICE simulations the entrance and exit points are the same as in [3]. In the graphs of Fig. 6 the simulation results concern the output voltage of the exit module, and that of the modules connected directly (from all possible directions) to the intersection module of every subsequent intersection in the path towards the entrance point. In the provided $V_{\mathrm{OUT}} - t$ graphs it is clear that the intersection module is trained due to an input signal coming from the neighboring module which belongs to the shortest path.

The graphs in Fig. 6 indicate the minimum required simulation time (nearly 80 min), which is still much shorter than the duration of the real experiments with Physarum. Moreover, we can note the difference in the frequency and amplitude of the output signals, which is attributed to the use of the south terminal at the intersections. The computed solution to the specific problem, according to the simulation results, includes the paths $\alpha 1$ and $\beta 1$ shown in Fig. 5(b), so it is partially in line with the experimental results in [3] in which the plasmodium always chose path $\alpha 2$. This difference is attributed to: i) the different computing strategy, which relied on filling the maze with plasmodium (i.e., providing the plasmodium with the complete information of the underlying maze); and ii) to the 2-D computing grid which results in the same Manhattan distance for routes $\alpha 1$ and $\alpha 2$. Therefore, using the proposed computing strategy it is equally possible to choose either route $\alpha 1$ or $\alpha 2$, even though in the biological experiment the Physarum could also move diagonally, something not supported by the computing medium.

In the following case study we use a different modeling strategy for route intersections which we evaluate based on biological experiments that are in line with the proposed computing methodology and concern placing the plasmodium to the entrance of the labyrinth and letting it scan the underlying network without providing any information about it.

V. Solution of a Circular Labyrinth

A. Biological Experiment

This case study concerns a more recent biological experiment with Physarum conducted by us in [5], in which the shape of the labyrinth is circular; the plasmodium is initially placed at the exterior level of the labyrinth and it then moves while

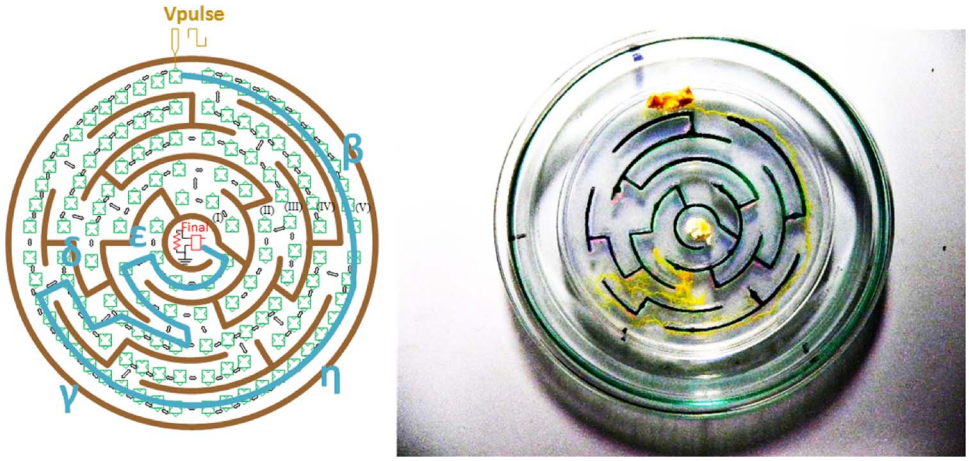


Fig. 11. Final computation result matching the data from the biological experiment.

searching for a path to the center of it, where the nutrient source is found. In this experimental setup the Physarum is following gradients of attractants. The searching strategy of Physarum in this case is absolutely in line with the proposed computing method. The experimental setup is shown in Fig. 7(a), whereas Fig. 7(b) shows the path approximated by the Physarum.

B. Electronic SPICE Modeling and Simulation

Similar to the square labyrinth case study, we designed a proper HW computing medium, shown in Fig. 8(a), where we mapped the target problem space by interconnecting properly the circuit modules. Due to the circular shape of this labyrinth, we defined five consecutive circular sectors with different radius, named I through V (counting from the sector closer to the center).

The walls of the labyrinth were placed between the circular sectors and the routes were created by connecting the modules according to the floor plan of the experimental labyrinth, shown in Fig. 8(b). The latter, for readability reasons, also indicates the end/start point of the labyrinth as well as all the alternative solution paths. Special attention was paid to the intersection wiring. In fact, due to the absence of an intermediate intersection module between two circular sectors, unlike in the square labyrinth, this time intersection modeling required using two couples of circuit modules, one from each of the circular sectors that join at the intersection, as shown in Fig. 9(a). In every such couple of modules belonging to the same sector, there is one circuit module which propagates the signal to the next sector and another one which receives a signal from it, thus modeling a bidirectional signal flow between every two joining sectors.

We again applied a sequence of 2-s-wide $\{V_o, V_p, T\} = \{-8\ \text{V}, 0.1\ \text{V}, 7\ \text{s}\}$ negative near-resonant square shaped voltage pulses to the input of the slime mould electronic module at the entrance (start point) of the labyrinth. During simulation, the modules belonging to almost all the alternative paths were trained, and the study of the points of interest [see simulation results in Fig. 10(a)–(c)] gave the shortest path consisting of the sections α, δ, and ϵ; i.e., a solution which is in line with the results in [5].

In laboratory experiment, the slime mould not rarely chooses slightly longer than shortest paths, or sometime average between shortest and longest. This is because living creatures are geared up towards efficiency not absolutely optimality [35]. The path chosen is an optimal in terms of energy already invested in the path search and envisaged economy in changing the adopted route. Also, it is not unusual for a slime mould to choose average path and only if the protoplasmic network is disrupted then re-grow the path towards a more optimal solution [36].

Consequently, in order to achieve the best possible modeling approach to the Physarum foraging behavior, while meeting the requirement for inclusion of natural noise in its environment, we placed a number of random noise voltage sources in all intersections. Fig. 9 summarizes all the possible types of connection between the circuit modules in the 2-D circular grid, where noise generators are included. Such sources of random noise may either slow down or prevent the expansion of the vascular network of the plasmodium towards a specific direction. This enhancement of the circuit model was proved essential to achieve the results given in the majority of the biological experiments. As shown in Fig. 9(b) and (c), there are two types of connection for the additional noise voltage sources; i.e., between two modules belonging to the same sector, or between modules joining two subsequent sectors at an intersection. In both cases we use an additional unidirectional module between the input terminal of the neighboring module and the noise generator source, to "filter" the high cumulative output voltage produced by the bidirectional composite circuit modules. In fact, the inter-sector noise generator required one more unidirectional module to function properly due to the problem caused while using the south terminal as input (explained in the previous section). During every simulation, a different file with random values was used for each noise voltage source.

After experimentation with the noise generators, we finally achieved simulation results which are in line with the experimental results [5]. More specifically, we preferably placed high noise in the routes α, $\sigma\tau$, and ζ, compared to the noise placed in the rest of the alternative paths. So, according to simulation results in Fig. 10(d)–(f), the computation resulted in the path $\beta, \eta, \gamma, \delta, \epsilon$, shown in Fig. 11. The simulation results concerning

TABLE I
SPICE NETLIST OF SLIME MOULD BEHAVIORAL CIRCUIT MODEL

```
.subckt RLCM In Out1 Out2 PARAMS:
*Memristor model parameters' values
+rmin=100 rmax=350 rinit=100 alpha=10000
beta=0.5 gamma=0.1 VtR=10 VtL=-4.5 yo=0.0001
+m=90 fo=0.55 Lo=5
******Stage (a)******
Rin In 2 0.1
Lin 2 RLCM_C 4
Cin RLCM_C 0 0.31
******Memristor model******
Gr 0 Memr value=dr_dt(V(RLCM_Output)-V(0))*
(st_f(V(RLCM_Output)-V(0))*
+st_f(V(Memr)-rmin)+st_f(-(V(RLCM_Output)-V(0)))
* st_f(rmax-V(Memr))
Cr Memr 0 1 IC={rinit}
Raux Memr 0 1E12
Gpm RLCM_Output 0 value={(V(RLCM_Output)-V(0))/
((fo*exp(2*L(V(Memr))))/L(V(Memr)))}
Rpm RLCM_Output 0 {1000*((fo*exp(2*Lo))/Lo)}
.func dr_dt(y)={-alpha*((y-VtL)/(gamma+abs(y-VtL)))
*st_f(-y+VtL)-beta*y*st_f(y-VtL)*
+st_f(-y+VtR)-alpha*((y-VtR)/(gamma+abs(y-VtR)))
*st_f(y-VtR)}
.func st_f(y)={1/(exp(-y/yo)+1)}
.func L(y)={Lo-Lo*m/y}
******Stage (b)******
Rcutter RLCM_C RLCM_Output 1
RDiode RLCM_Output 0 5.827E+9
D1 RLCM_Output 0 1N4148
******Stage (c)******
Vthres OpAmpInv 0 DC -4.5
********OpAmp********
c1op op11 op12 8.661E-12
c2op op6 op7 30.00E-12
dcop OpAmpOut op53 dy
deop op54 OpAmpOut dy
dlpop op90 op91 dx
dlnop op92 op90 dx
dpop DCNeg DCPos dx
egndop op99 0 poly(2),(DCPos,0),(DCNeg,0) 0 0.5
0.5
fbop op7 op99 poly(5) vbop vcop veop vlpop
vlnop 0 10.61E6 -1E3 1E3 10E6 -10E6
gaop op6 0 op11 op12 188.5E-6
gcmop 0 op6 op10 op99 5.961E-9
ieeop op10 DCNeg dc 15.16E-6
hlimop op90 0 vlimop 1K
q1op op11 OpAmpInv op13 qx
q2op op12 RLCM_Output op14 qx
r2_Op_Amp op6 op9 100.0E3
rc1op DCPos op11 5.305E3
rc2op DCPos op12 5.305E3
re1op op13 op10 1.836E3
re2op op14 op10 1.836E3
reeop op10 op99 13.19E6
ro1op op8 OpAmpOut 50
ro2op op7 op99 100
rpop DCPos DCNeg 18.16E3
vbop op9 0 dc 0
vcop DCPos op53 dc 1
veop op54 DCNeg dc 1
vlimop op7 op8 dc 0
vlpop op91 0 dc 40
vlnop 0 op92 dc 40
.model dx D(Is=800.0E-18 Rs=1)
.model dy D(Is=800.00E-18 Rs=1m Cjo=10p)
.model qx NPN(Is=800.0E-18 Bf=93.75)
Vpos DCPos 0 DC 10
Vneg DCNeg 0 DC -10
******Stage (d)******
Rprebuffer OpAmpOut Cutter 1k
RDthres1 Cutter Thres1 5.827E+9
Dthres1 Cutter Thres1 1N4148
Vthres1 Thres1 0 DC -0.6
RDThres2 Thres2 Cutter 5.827E+9
DThres2 Thres2 Cutter 1N4148
Vthres2 Thres2 0 DC -3.3
******Stage (e)******
Rfinal Cutter 0 10k
EOut Out1 Out2 Cutter 0 1.173
********Diode********
.MODEL 1N4148 D
+ IS = 4.352E-9
+ N = 1.906
+ BV = 110
+ IBV = 0.0001
+ RS = 0.6458
+ CJO = 7.048E-13
+ VJ = 0.869
+ M = 0.03
+ FC = 0.5
+ TT = 3.48E-9
.ends
```

the output voltage of the destination module and of the modules in the intersections of interest, shown in Fig. 10, highlight the differences between the two approaches. According to Fig. 10(a) and (d), computation finished successfully in both cases, even though without any noise generators it required much less time to reach the destination via the shortest path α, δ, ϵ, shown previously in Fig. 8(b). Next, attention is paid to intersection 1 to figure out the direction of the signal that first reaches this point. As shown in Fig. 10(b) and (e), either when subject to noise or not, the signal arrived through the route in section δ and not from $\sigma\tau$. Finally, in Fig. 10(c) and (f) we focus on the three intersections [highlighted in Fig. 8(b)] from

which the signal could pass from sector V to IV. The measured voltages concern the route sections δ, $\sigma\tau$, and ζ. Without the presence of noise generators, the signal appears from route δ before it does from route ζ, so obviously it followed the route α and not η and γ, as expected. Otherwise, it would have also appeared previously from route ζ as well. Similarly, when the route α contains high noise, the signal enters sector IV from route δ, which means that it followed routes β, η, and γ.

Consequently, this new attribute of the computing circuit model concerning natural noise, enables a more naturally controlled propagation, which could lead to significant findings and improved HW-based knowledge on the behavioral characteristics of slime mould.

VI. Conclusion

In this paper, we presented a hardware-based maze-solving approach via an electronic model of the oscillatory internal motion mechanism of plasmodium. The efficiency and generality of the proposed electronic circuit was validated through SPICE-level simulations compared with data from two different biological experiments, namely enhancing of Physarum's protoplasmic tubes along shortest path [3], and chemo-tactic growth of slime mould [5]. The proposed circuit is operated in the region of maximum adaptation (learning), so every module adjusts its function depending on the characteristics of the input waveform, whereas network expansion corresponds to gradual training of subsequent interconnected modules. The introduction of inherent noise abilities to the equivalent biological circuit resulted in several (including the optimal) maze solutions which is closer to non-deterministic real organisms' behavior [3], [5]. Future work could include modifications to the proposed circuit modules so as to support also the shrinking stage of the vascular network of the plasmodium after the shortest possible interconnection network has been computed. The series connection of distributed identical modules with circuit parameters adjusted for specific transmission characteristics resembles the distributed element model of transmission lines. Therefore, the proposed here circuit could be optimized to minimize circuit overhead by stages b–e through optimal parameter value selection for perfect signal transmission and reflection to achieve self-reinforcement of the solution paths. Other extensions of the proposed work could include the circuit modeling of the behavioral characteristics of plasmodium, demonstrated in a variety of biological experiments concerning the solution of several other problems such as the traveling salesman or the execution of logic computations.

Appendix

See Table I.

References

[1] A. Adamatzky, Ed., *Physarum Machines: Making Computers From Slime Mould*, Singapore: World Scientific, 2010.

[2] X. Zhang, A. Adamatzky, F. T. S. Chan, Y. Deng, H. Yang, X.-S. Yang, M.-A. I. Tsompanas, G. Ch. Sirakoulis, and S. Mahadevan, "A biologically inspired network design model," *Scientific Reports*, vol. 5, pp. 455–467, 2015.

[3] T. Nakagaki, H. Yamada, and A. Toth, "Maze-solving by an amoeboid organism," *Nature*, vol. 407, no. 6803, p. 470, Sep. 2000.

[4] A. Tero, S. Takagi, T. Saigusa, K. Ito, D. P. Bebber, M. D. Fricker, K. Yumiki, R. Kobayashi, and T. Nakagaki, "Rules for biologically inspired adaptive network design," *Science*, vol. 327, no. 5964, pp. 439–442, Jan. 2010.

[5] A. Adamatzky, "Slime mold solves maze in one pass, assisted by gradient of chemo-attractants," *IEEE Trans. NanoBiosci.*, vol. 11, no. 2, pp. 131–134, Jun. 2012.

[6] A. Schumann, K. Pancerz, A. Adamatzky, and M. Grube, "Bio-inspired game theory: The case of physarum polycephalum," in *Proc. 8th Int. Conf. Bioinspired Inf. Commun. Technol., ser. BICT*, 2014, pp. 9–16, ICST, Brussels, Belgium, Belgium: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

[7] M. Aono and M. Hara, "Spontaneous deadlock breaking on amoeba-based neurocomputer," *Biosystems*, vol. 91, no. 1, pp. 83–93, 2008.

[8] L. Zhu, M. Aono, S.-J. Kim, and M. Hara, "Amoeba-based computing for traveling salesman problem: Long-term correlations between spatially separated individual cells of physarum polycephalum," *Biosystems*, vol. 112, no. 1, pp. 1–10, 2013.

[9] E. Gale, A. Adamatzky, and B. Lacy Costello, "Slime mould memristors," *BioNanoScience*, vol. 5, no. 1, pp. 1–8, 2014.

[10] T. Saigusa, A. Tero, T. Nakagaki, and Y. Kuramoto, "Amoebae anticipate periodic events," *Phys. Rev. Lett.*, vol. 100, Jan. 2008, Art. no. 018101.

[11] F. Traversa, Y. Pershin, and M. Di Ventra, "Memory models of adaptive behavior," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 9, pp. 1437–1448, Sep. 2013.

[12] L. Chua, "If it's pinched it's a memristor," *Semicond. Sci. Technol.*, vol. 29, no. 10, 2014, Art. no. 104001.

[13] I. Vourkas and G. Sirakoulis, *Memristor-Based Nanoelectronic Computing Circuits and Architectures: Foreword by Leon Chua*, ser. Emergence, Complexity and Computation, New York, NY, USA: Springer-Verlag, 2015.

[14] M. Ziegler, K. Ochs, M. Hansen, and H. Kohlstedt, "An electronic implementation of amoeba anticipation," *Appl. Phys. A*, vol. 114, no. 2, pp. 565–570, 2014.

[15] C. E. Shannon, "Presentation of a maze-solving machine," in *Proc. 8th Conf. Josiah Macy Jr. Found. (Cybernetics)*, 1951, pp. 173–180.

[16] O. Steinbock, Á. Tóth, and K. Showalter, "Navigating complex labyrinths: Optimal paths from chemical waves," *Science*, vol. 267, no. 5199, pp. 868–868, 1995.

[17] K. Agladze, N. Magome, R. Aliev, T. Yamaguchi, and K. Yoshikawa, "Finding the optimal path with the aid of chemical wave," *Physica D: Nonlinear Phenom.*, vol. 106, no. 3, pp. 247–254, 1997.

[18] N. Rambidi and D. Yakovenchuck, "Finding paths in a labyrinth based on reaction-diffusion media," *BioSystems*, vol. 51, no. 2, pp. 67–72, 1999.

[19] N. Rambidi and D. Yakovenchuk, "Chemical reaction-diffusion implementation of finding the shortest paths in a labyrinth," *Phys. Rev. E*, vol. 63, no. 2, 2001, Art. no. 026607.

[20] A. Adamatzky and B. de Lacy Costello, "Collision-free path planning in the Belousov–Zhabotinsky medium assisted by a cellular automaton," *Naturwissenschaften*, vol. 89, no. 10, pp. 474–478, 2002.

[21] D. R. Reyes, M. M. Ghanem, G. M. Whitesides, and A. Manz, "Glow discharge in microfluidic chips for visible analog computing," *Lab on a Chip*, vol. 2, no. 2, pp. 113–116, 2002.

[22] A. E. Dubinov, A. N. Maksimov, M. S. Mironenko, N. A. Pylayev, and V. D. Selemir, "Glow discharge based device for solving mazes," *Phys. Plasmas (1994-Present)*, vol. 21, no. 9, 2014, Art. no. 093503.

[23] I. Lagzi, S. Soh, P. J. Wesson, K. P. Browne, and B. A. Grzybowski, "Maze solving by chemotactic droplets," *J. Amer. Chem. Soc.*, vol. 132, no. 4, pp. 1198–1199, 2010.

[24] J. Cejkova, M. Novak, F. Stepanek, and M. M. Hanczyc, "Dynamics of chemotactic droplets in salt concentration gradients," *Langmuir*, vol. 30, no. 40, pp. 11 937–11 944, 2014.

[25] A. Reynolds, "Maze-solving by chemotaxis," *Phys. Rev. E*, vol. 81, no. 6, 2010, Art. no. 062901.

[26] P. Lovass, M. Branicki, R. Tóth, A. Braun, K. Suzuno, D. Ueyama, and I. Lagzi, "Maze solving using temperature-induced Marangoni flow," *RSC Advances*, vol. 5, no. 60, pp. 48 563–48 568, 2015.

[27] V. Ricigliano, J. Chitaman, J. Tong, A. Adamatzky, and D. G. Howarth, "Plant hairy root cultures as plasmodium modulators of the slime mold emergent computing substrate physarum polycephalum," *Frontiers in Microbiol.*, vol. 6, Jul. 2015, Art. no. 720.

[28] V. Ntinas, I. Vourkas, and G. Sirakoulis, "LC filters with enhanced memristive damping," in *Proc. IEEE ISCAS*, May 2015, pp. 2664–2667.

[29] Y. V. Pershin, and M. Di Ventra, "Solving mazes with memristors: A massively parallel approach," *Phys. Rev. E*, vol. 84, no. 4, 2011, Art. no. 046703.

[30] Z. Ye, S. H. M. Wu, and T. Prodromakis, " Computing shortest paths in 2D and 3D memristive networks," in *Memristor Networks*, New York, NY, USA: Springer-Verlag, 2014, pp. 537–552.
[31] I. Vourkas, A. Batsos, and G. C. Sirakoulis, "Spice modeling of nonlinear memristive behavior," *Int. J. Circuit Theory Appl.*, vol. 43, no. 5, pp. 553–565, 2015.
[32] A. Adamatzky, "Slime mould electronic oscillators," *Microelectron. Eng.*, vol. 124, pp. 58–65, 2014.
[33] Y. V. Pershin, S. La Fontaine, and M. Di Ventra, "Memristive model of amoeba learning," *Phys. Rev. E*, vol. 80, Aug. 2009, Art. no. 021926.
[34] D. B. Strukov, G. S. Snider, D. R. Stewart, and R. S. Williams, "The missing memristor found," *Nature*, vol. 453, no. 7191, pp. 80–83, 2008.
[35] A. Adamatzky, "Route 20, autobahn 7, and slime mold: Approximating the longest roads in USA and Germany with slime mold on 3-D terrains," *IEEE Trans. Cybernetics*, vol. 44, no. 1, pp. 126–136, 2014.
[36] C. R. Reid and M. Beekman, "Solving the towers of hanoi-how an amoeboid organism efficiently constructs transport networks," *J. Exper. Biol.*, vol. 216, no. 9, pp. 1546–1551, 2013.

Vasileios Ntinas received the M.Eng. degree in electrical and computer engineering from the Democritus University of Thrace (DUTh), Xanthi, Greece, in 2015. He is currently pursuing the M.Sc. degree at the Electrical and Computer Engineering Department of DUTh. His thesis emphasis is on the design and simulation of bio-inspired memristor-based computing circuits and systems.

Ioannis Vourkas (S'12–M'16) received the M.Eng. (Diploma) and the Ph.D. degrees in electrical and computer engineering from the Democritus University of Thrace (DUTh), Xanthi, Greece, in 2008 and 2014, respectively.

Currently, he is a Post-Doctoral Researcher with the Department of Electrical Engineering of Pontificia Universidad Católica de Chile (PUC), Santiago, Chile. In 2006, he was summer-intern in the Cultural and Educational Technology Institute in Xanthi, Greece. In 2009, he worked via the European vocational training program "Leonardo Da Vinci" in the Institute of Industrial and Control Engineering of the Polytechnic University of Catalonia (UPC), Barcelona, Spain. In 2012, he was with the organizing committee of the 2012 ACRI Conference, where he met the "inventor" of memristor, Prof. Chua. His current research emphasis is on novel nanoelectronic circuits and architectures comprising memristors. His research interests include the design of nanoelectronic integrated circuits, unconventional computing, software, and hardware aspects of parallel complex computational (bioinspired) circuits and systems, cellular automata and their applications. In these areas, so far he is main author of one book, one book chapter, and more than 30 scientific journal/ conference papers.

Dr. Vourkas is distinguished as the 2014 Best Ph.D. Student in ECE-DUTh. He was also invited Tutorial speaker in the 2015 IEEE International Conference on Electronics, Circuits, and Systems (ICECS) held in Cairo, Egypt. He was a scholar of the Greek Bodossaki foundation from 2011 to 2014.

Georgios Ch. Sirakoulis (M'95) received the Dipl. Eng. and Ph.D. degrees in electrical and computer engineering from Democritus University of Thrace (DUTh), Xanthi, Greece, in 1996 and 2001, respectively.

Since 2008, he has been an Associate Professor with tenure in the Electrical and Computer Engineering Department, DUTh. He serves as Associate Editor in *Microelectronics Journal*, *Integration*, *The VLSI Journal*, *International Journal of Parallel, Emergent and Distributed Systems*, *Journal of Cellular Automata*, *Journal of Applied Mathematics* and *Recent Advances on Electrical & Electronic Engineering*. He has published more than 200 technical papers and 10 guest-editorials; he is co-editor of five books and co-author of 15 book chapters. He is EUROPRACTICE representative for DUTh, and he has served as a member at the EU IDEAS programme. He has participated as a principal investigator in more than 20 scientific programmes and projects funded from the Greek Government and Industry as well as the European Commission. He was general co-chair of ACRI 2014 Conference in Krakow, Poland, and of several other conferences like ACRI 2012, SFHMMY 2012, etc. and Special Sessions in IEEE ICECS 2013, PDP 2014, ICCSA 2014, PDP 2015, PCI 2015, IEEE NANO 2015, etc.

Dr. Sirakoulis received a prize of distinction from the Technical Chamber of Greece (TEE) for his Diploma Thesis in 1996, and he was also founding member and Vice President of the IEEE Student Branch of Thrace for the period 2000–2001. He is a member of IEEE, of IEEE Circuits and Systems, of IEEE Computer Society, of the Association of Computing Machinery (ACM), of the International Society for Computational Biology (ISCB), and a member of TEE. His current research emphasis is on emergent circuits and devices, modern electronic models and architectures with memristors, Green and unconventional computing, Cellular Automata theory and applications, complex systems, bio-inspired computation/bio-computation, modeling and simulation.

Andrew I. Adamatzky is Professor in the Department of Computer Science and Director of the Unconventional Computing Centre, University of the West of England, Bristol, UK. He does research in reaction-diffusion computing, cellular automata, physarum computing, massive parallel computation, applied mathematics, collective intelligence and robotics, bionics, computational psychology, nonlinear science, novel hardware, and future and emergent computation.