Design of single-electron “slime-mold” circuit and its application to solving optimal path planning problem

Yuya Shinde¹ and Takahide Oya¹a)

¹ Graduate School of Engineering, Yokohama National University
79-5 Tokiwadai, Hodogaya-ku, Yokohama 240-8501, Japan

a) t-oya@ynu.ac.jp

Received May 31, 2013; Revised September 10, 2013; Published January 1, 2014

Abstract: We propose a single-electron “slime mold” circuit that can find the shortest path of a maze based on the behaviors of slime molds. The circuit consists of a single-electron reaction-diffusion (SE-RD) circuit and logic gate circuits. An SE-RD circuit can be used to represent the dilatation behaviors of slime molds. In this study, a slime mold model that was originally developed for cellular automata is used for the logic gate circuit. We used a Monte Carlo simulation to confirm that the proposed circuit can operate like slime molds. Moreover, it was able to find the optimal path of any maze.

Key Words: single-electron circuit, slime mold, biomimetic information processing, nonlinear circuit

1. Introduction
Some researchers have recently been focusing on the unique behavior of a certain type of unicellular slime mold because it has the capability to solve mazes despite being a unicellular animal [1], which is very interesting. In fact, one research group (Nakagaki et al.) won the Ig Nobel Prize in Cognitive Science in 2008 [2] because of the impact of their research. The process of solving maze problems using slime molds has two phases. In the first phase, the slime mold spreads throughout the maze by taking all possible routes. The second phase is the shrinking phase. If food is placed at only the start and goal points of the maze, the slime will recede at the dead end points with no food in the maze, and finally, it will connect only the start and goal points using the shortest distance in order to efficiently take in the food. Therefore, the slime mold can discover the optimal answer to the problem of how to most efficiently connect two places that contain food [1]. Such a unique behavior in the natural world has also provided researchers in biomimetic engineering with new ideas or hints for how to materialize novel robots, electrical devices, mathematical models, and so on.

We designed single-electron (SE) “slime mold” circuits for this study that mimic the behavior of unicellular slime molds for use as novel biomimetic SE analog circuits. The SE circuit is an electronic circuit designed to manipulate electronic functions by controlling the transport of individual electrons [3]. It has been attracting attention for use as a novel nonlinear device and a candidate for next generation electric devices because it can ensure there is a higher density with an ultra low power.
consumption because it has unit elements on the nanometer scale, unlike conventional devices. Various useful SE devices have previously been proposed, including the SE transistor (e.g., [4]), the SE pump (e.g., [5]), the SE memory (e.g., [6]), and the majority logic circuit (e.g., [7]). We proposed an SE reaction-diffusion (RD) circuit [8] for use as a biomimetic or nature-inspired SE circuit. An original RD system is a chemical system where the chemical reactions and material diffusion coexist in a non-equilibrium state. It produces orderly spatiotemporal patterns of a chemical concentration called “dissipative structures.” Such dynamics have already been expressed as mathematical or computing models by many researchers. Our SE-RD circuit was also based on these RD models. These models can express a certain type of useful natural phenomena and even the behavior of a creature. Therefore, we have tried to design unique systems on the basis of our SE-RD circuit for use as applications. As a result, we have already succeeded in designing one and have proposed a device for computing a Voronoi diagram [9], nature-inspired cellular automata for dendritic pattern formation [10], and a collision-based computing circuit [11]. The SE slime-mold we propose in this study is also based on the SE-RD circuit, and also introduces a slime-mold model for cellular automata. We designed the circuit and evaluated its operations by using the Monte-Carlo computer simulation method. Moreover, we observed how the circuit operates when used for solving the optimal path planning problem.

2. Information processing for maze problem by using slime mold

We are focusing on the information-processing ability of a “slime mold” as reported in [1] in this study. The plasmodium of the slime mold Physarum polycephalum, which is a large amoeba-like cell consisting of a dendritic network of tube-like structures (pseudopodia), was studied in their report. It basically changes its shape when it is moving or searching for food. When food is placed at two or more different points and the slime mold finds it, it will put out pseudopodia that connect the two or more food sources to absorb them. As a result of this behavior, the formed pseudopodia show the shortest paths among the food sources. Therefore, it can be assumed that this simple organism has the ability to find the minimum-length solution among two or more points in a maze.

This unique behavior of slime mold can be expressed using mathematical models. For example, Ikebe et al. proposed a virtual slime-mold model for computational cellular automata in order to solve a multi-path maze problem. There are two key points for expressing this distinctive behavior in terms of the topology preservation of slime mold. One is the “dilatation” behavior, and the other is the “contraction” behavior. Moreover, such behaviors can be considered as “scanning for food” using dilatation, “securing food,” and “effectively obtaining food” by contracting its body. These three states are very important for solving the multi-path maze problem. The virtual slime mold designed by Ikebe et al. [12] has these states as its cellular automata. For operations, the small virtual slime mold body is put on the inside of a maze (a start point) in the initial state. As a first step, which is scanning for food for preserving one’s own topology, the slime mold spreads throughout the inside of the maze and looks for food that is placed at a given goal point. This dilatation behavior can be easily mimicked by using an RD system for example. The information that the slime mold has come into contact with food will spread throughout the slime mold during the secure food phase. Then, for effectively obtaining food, the slime mold mutates its body in order to efficiently obtain the food (The slime mold indicates the shortest path in the maze). Mimicking this contraction behavior is important for developing slime mold circuitry. Ikebe et al. proposed a unique contraction rule for a cellular automata to mimic this behavior [12]. They prepared three templates for the rule, as shown in Fig. 1(a). The cell state was binary. Where a logical 1 represents an object (the slime mold body) and a 0 represents nothing or space (corridors). Here, the corridor cells of the maze at which the slime mold does not arrive are assumed to be logical 0. On the other hand, the corridor cell changes its logical state from 0 to 1, when the slime mold arrives. The blank cells in each template are allowed to have any state (1 or 0). When the contraction rule is executed, the blank cells are omitted to calculate the next state. Each template represented an inward contraction that was slanted and convex. Furthermore, they did not disrupt the topology. If the templates matched, the cell state changed from 1 to 0. They shrunk the objects from four different directions by rotating the templates.
90 degrees for each step. All the templates are compared with the object at the same timing in each step, and the next state is then decided. Unlike in the case of a conventional isotropic rule, the form of an object did not disappear (Fig. 1(b)). The topologies of objects without a hole converge onto one point. When the slime mold is used, it automatically chooses the above-mentioned states on its own. Thus, the multi-path maze problem can be skillfully solved when using this virtual slime mold operation.

3. Single-electron circuits for “slime mold” circuit

The designed circuit must be able to perform both the “dilatation” and “contraction” operations in order to be used as a single-electron slime-mold circuit. For the dilation operation, we use our SE-RD circuit as a component of the slime-mold circuit because the circuit can express the propagation of excited waves like that from the dilatation of a slime mold [8]. Figure 2 shows the circuit configuration.
of the original SE-RD circuit. The main component is an SE oscillator that consists of a tunneling junction, \( C_j \), and a high resistance, \( R \), connected in a series at a node and biased by a positive voltage, \( V_d \), or a negative one, \(-V_d\). It has threshold value for electrons to tunnel. When the electron tunneling occurs in a \( V_d\)-biased SE oscillator, for example, the voltage \( V_{node} \) of the node of the oscillator suddenly changes its voltage from positive to negative. This sudden change of voltage triggers other electron tunnelings in adjacent oscillators. This behavior is indispensable for imitating RD systems [8]. As a result, the SE-RD circuit can generate dilatation-like voltage waves caused by occurrence of an electron tunneling in each SE oscillator in the SE-RD circuit, as shown in Fig. 3. We can assume that the occurrence of electron tunnelings represents logical 1, and that the nonoccurrence represents 0. In contrast, the SE-RD cannot express the contraction operation of the slime-mold circuit. Therefore, additional circuits are required to design the target circuit. We designed additional circuits that are based on the virtual slime-mold model [12] for this study, and tried to combine the SE-RD and the additional circuits in order to construct the SE slime-mold circuit. The three templates shown in Fig. 1(a) are required to express the virtual slime mold. Moreover, each template must be rotated every 90°, as shown in Fig. 1(b). The upper template of Fig. 1(a) can be represented as the logical equation

\[
\neg X_{c,t+1} = X_{c,t} \cdot X_{s,t} \cdot \neg X_{nw,t} \cdot \neg X_{n,t} \cdot \neg X_{w,t} \cdot X_{e,t} \cdot X_{se,t},
\]

the middle can also be represented as

\[
\neg X_{c,t+1} = X_{c,t} \cdot X_{s,t} \cdot \neg X_{nw,t} \cdot \neg X_{n,t} \cdot \neg X_{ne,t} \cdot \neg X_{w,t} \cdot \neg X_{c,t},
\]

and the bottom can also be represented as

\[
\neg X_{c,t+1} = X_{c,t} \cdot X_{s,t} \cdot \neg X_{nw,t} \cdot \neg X_{n,t} \cdot \neg X_{ne,t} \cdot X_{sw,t} \cdot X_{se,t}.
\]
Fig. 4. Circuit configuration of seven-input SE-NAND circuit on basis of SE trap.

Here, $X_c$ is the logical state (1 or 0) of the center cell of each template, $t$ is the current time, and $t + 1$ represents the next step. $X_n, X_w$, and so on are the neighbors of the center cell, e.g., $n$ means that the cell is placed on the north side of the center cell. We considered using logic gates that have seven input terminals, i.e., we use arrayed seven-input SE-NAND gates, to ensure the templates can be used as SE circuits. Each template must have the same number of gate circuits as SE oscillators, and we assume each gate is placed under each oscillator. In our circuit, the variable $X$ is assumed to be the $V_{\text{node}}$ of the SE oscillator.

Figure 4 shows a gate circuit that consists of an SE trap (two identical tunneling junctions, $C_J$, connected in a series, a bias capacitor, $C_L$, and a bias voltage, $V_D$) with seven input capacitors ($C_{\text{in}}$) and an output capacitor ($C_o$). The gate circuit is the modified circuit that we already reported on in [7, 13]. The input signals $V_{\text{in}1} \sim V_{\text{in}7}$ for the gate circuit are the $V_{\text{node}}$s of the SE oscillators in the SE-RD circuit. For Eq. (1), for example, the inputs can be represented as

$$
\begin{pmatrix}
V_{\text{in}1} \\
V_{\text{in}2} \\
V_{\text{in}3} \\
V_{\text{in}4} \\
V_{\text{in}5} \\
V_{\text{in}6} \\
V_{\text{in}7}
\end{pmatrix} =
\begin{pmatrix}
V_{\text{node},c} \\
V_{\text{node},s} \\
-V_{\text{node},nw} \\
-V_{\text{node},n} \\
-V_{\text{node},ne} \\
V_{\text{node},sw} \\
V_{\text{node},se}
\end{pmatrix}.
$$

(4)

We prepared and added templates rotated 90°, 180°, and 270° for each template as new templates for easy demonstration to express the rotation of the templates for every 90°. As a result, the circuit can express the behavior of the virtual slime mold by using the SE-RD circuit and 12 $(3 \times 4)$ templates.

4. Simulation results

We conducted a computer simulation based on the Monte Carlo method to test the operation of our circuit. Figures 5 and 6 show the simulation results. The device used in the simulation had $256 \times 256$ elements in each layer and calculated the simple maze (corridor) problem, and we assumed that our circuit had an output layer. Ideally, the output layer consists of arrayed 12-input OR gate circuits connected to each NAND gate in each template. Here, the output signals were calculated from the outputs of 12 templates for an easy demonstration using the simulator. Corridor and wall parts were required to represent the maze on the circuit. As explained in [9], by changing the bias voltage $V_d$ for an SE oscillator, the oscillator can act as both the corridor that allows the voltage wave to propagate and the wall that denies. If the bias voltage $V_d$ is set to near the threshold in a subthreshold state, then the oscillator will act as a corridor. On the other hand, if the $V_d$ is set to a low enough voltage compared to the threshold, the oscillator will act as a wall. For the tests, we set the temperature at 0 K and used the circuit parameters $C_J = 20 \text{ aF}$, $C = 2.2 \text{ aF}$, $R = 25 \text{ M}\Omega$, and $V_d = 6.1 \text{ mV}$ for the corridors, and $V_d = 5.0 \text{ mV}$ for the walls for the SE oscillators, and $C_J = 20 \text{ aF}$, $C_L = 6 \text{ aF}$, $C_{\text{in}} = 2.2 \text{ aF}$, and $V_D = 20 \text{ mV}$ for the SE-NAND circuits. In [12], the virtual slime-mold system generated a wave (dilatation) from the start point, and executed the contraction operation by using the three templates after the wave spread all over the maze in order to correctly calculate the optimal path planning problem. In this study, the SE-RD circuit was used for the dilatation operation, and
the 12 templates were used for the contraction operation. In the virtual slime-mold model [12], data of logical 1 caused by slime mold arriving were kept until the phase changed from the dilatation operation to contraction one. Therefore, this memory function is also required for our circuit. For simplicity in solving the maze problem using our circuit, we assumed that there were additional $256 \times 256$ SE traps that contained an input and an output capacitor on another layer to record data of the occurrence of the electron tunneling (logical 1) in each SE oscillator of the SE-RD circuit. The contraction circuit (12 templates) solved the maze problem by using the memorized data after the voltage waves spread all over the maze. These operations were based on the demonstrated operations shown in Fig. 1(b). The additional SE traps for memory use can be omitted by changing the value of $R$ in each SE oscillator in the SE-RD circuit to larger values. Figure 5 shows some snapshots of the density of the node voltages on the SE-RD device layer. A bright color means the node voltage was high and a dark color means the node voltage was low. Figure 6 shows the voltages on the output layer. In this simulation, we triggered an oscillator on the SE-RD layer at the upper left for use as a planar point for the dilatation of the virtual slime mold. The voltage waves traveled within a corridor like that for dilatation. In the output layer, the elements changed their node voltages like that for the contraction of the virtual slime mold when a NAND circuit on any template outputted a signal. This implies that our circuit has the ability to solve the maze problem just like slime molds do.

Fig. 5. Dilatation operation in SE-RD layer of device shown as snapshots of four time steps.

Fig. 6. Contraction operation in output layer of device shown as snapshots for four time steps.
Finally, we tested the operation of the circuit using another maze with two paths as a demonstration. Figures 7 and 8 show the simulation results. Figure 7 shows snapshots of the node voltages on the SE-RD device layer. Figure 8 shows the final state of the voltages on the output layer. As a result, the device also showed solutions as an output.

5. Conclusion

We proposed a single-electron “slime mold” circuit and its application to one unconventional analog information processing system, i.e., the optimal path planning problem. The proposed circuit consists of three kinds of layers. The first layer is an SE-RD circuit in which voltage waves are generated and travel like the contraction behaviors of slime molds, the second layer has 12 templates that consist of arrayed seven-input SE-NAND gates connecting to the first layer on the basis of the theory of the virtual slime mold model [12] for mimicking the dilatation behaviors of slime molds, and the third layer is an output layer that consists of arrayed 12-input OR gates connected to each template in the second layer. The results from a demonstration done by computer simulation confirmed that our circuit can be used as a “slime mold” circuit.

In this demonstration, our circuit did not solve the (difficult) optimal path planning problem in the strict sense. However, the proposed circuit was based on the algorithm as described in [12]. The algorithm can solve the optimal path planning problem correctly. Therefore, we believe that our circuit also can solve the problem. For that, our circuit must have an additional circuit. Because,
according to [12], the system requires data of collision points of the traveling waves on the maze to solve the problem. In contrast, our circuit has not had a function to memory the data of collision points so far. For this issue, we have a candidate of the answer. We had already proposed the unique SE circuit for computation of a Voronoi diagram as described in [9]. The circuit has the memory function for the data of collision points of the traveling waves. Therefore, our slime mold circuit must be able to solve the optimal path planning problem correctly as same as the virtual slime mold [12] by adding the memory circuit based on [9].

We also gave no consideration to the number of coupling capacitors between each layer in this paper. We now need to consider ways of coupling the unit circuits between the layers. However, coupling the capacitors or wires would occupy a huge amount of space within the device. We tested the operation of a device with $256 \times 256$ elements in each layer and 12 templates in the second layer, which would need at least 6,356,000 coupling capacitors in total. We prepared 12 templates in the second layer for mimicking the dilatation behavior for easy demonstration. If we can design a second layer using three templates for use in the same way as the virtual slime mold, we believe we could use 75% less coupling capacitors. Moreover, we must consider the influence of any fluctuation in the circuit parameters and the noise effects of the proposed circuit for effective operations. If the circuit has any fluctuation in the circuit parameters or is placed in a noisy environment, it will not be able to be correctly operated. However, we believe we can overcome these problems because we are now looking into a device-error-redundant SE circuit [14] and a thermal-noise-harnessing SE circuit [15], and have already confirmed such circuits can be correctly operated. We will therefore consider a simpler structure with additional functions in our future work.

Acknowledgments

This work was supported by a Grant-in-Aid for Young Scientists (B) (No. 24710149) and a Grantin-Aid for Scientific Research on Innovative Areas (No. 25110015) from the Japan Society for the Promotion of Science (JSPS).

References


