

Robust Soldier Crab Ball Gate

Yukio-Pegio Gunji

Yuta Nishiyama

Department of Earth and Planetary Sciences

Kobe University

Kobe 657-8501, Japan

Andrew Adamatzky

Unconventional Computing Centre

University of the West of England

Bristol, United Kingdom

Soldier crabs *Mictyris guinotae* exhibit pronounced swarming behavior. Swarms of the crabs are tolerant of perturbations. In computer models and laboratory experiments we demonstrate that swarms of soldier crabs can implement logical gates when placed in a geometrically constrained environment.

1. Introduction

All natural processes can be interpreted in terms of computations. To implement a logical gate in a chemical, physical, or biological spatially extended medium, Boolean variables must be assigned to disturbances, defects, or localizations traveling in the medium. These traveling patterns collide and the outcome of their collisions are converted into resultant logical operations. This is how collision-based computers work [1, 2]. Now, classical examples of experimental laboratory unconventional computing include the Belousov-Zhabotinsky (BZ) medium and the slime mold of *Physarum polycephalum*. In BZ, excitable medium logical variables are represented by excitation waves that interact with each other in the geometrically constrained substrate or “free-space” substrate [3–6]. Slime mold is capable of solving many computational problems, including maze and adaptive networks [7, 8]. In the case of ballistic computation [9], slime molds implement collision computation when two slimes are united or avoid each other dependent on the gradient of attractor and inhibitor. We previously suggested that the slime mold logical gate is robust against external perturbation [2, 10]. To expand the family of unconventional spatially extended computers, we studied the swarming behavior of soldier crabs *Mictyris guinotae* and found that compact propagating groups of crabs emerge and endure under noisy external stimulation. We speculated that swarms can behave similarly to billiard balls and thus implement basic circuits of collision-based computing. The results of our studies are presented in this paper.

2. Swarming of Soldier Crabs

Soldier crabs *Mictyris guinotae* inhabit flat lagoons and form huge colonies of several hundreds and sometimes hundreds of thousands of crabs. In field expeditions to Funaura Bay, Iriomote Island, Japan, we observed a peculiar wandering behavior of crabs. A front part of their swarm is driven by inherent turbulence. The arrangement of individuals is always changing. A single crab or a small group of crabs do not usually enter the water; however, a large swarm enters the water and crosses a lagoon without hesitation. The large swarm crossing the water consists of an active front and passive tail. The crabs in the tail simply follow the crabs at the front. We assumed there are two types of neighborhoods: one for positive interaction and another for monitoring and following flock-mates.

To implement these behaviors in a model, we introduce a number of potential transitions for each individual that can be employed to mutual anticipation [11]. Each individual has its own principal vector representing its own velocity, accompanied with a number of potential transitions P in a range restricted by angle α . The neighborhood has dual roles: the extended body of an individual for active interaction and the local space for passively monitoring other individuals.

Figure 1 shows a typical transition of our swarm model of which individuals are numbered from 1 to 3. First, velocity matching is applied to the principal vector by averaging all flock-mate velocities in the neighborhood, NM, which is defined here as a Moore neighborhood extended by next-nearest neighboring cells (Figure 1(a), left and center in top row). If targets of some potential transitions are overlapped at a particular cell (Figure 1(a) right in top row), the overlapping is counted as “popularity.” In Figure 1(a), left in center row, transitions from individuals 2 and 3 overlap at a pink cell, meaning that the pink cell has a popularity of 2. If some potential transitions reach highly popular sites beyond the threshold of popularity (defined here by 1), an individual moves to the site with the highest popularity. In Figure 1(a), center in center row, the individual 2 moves to the most popular cell. If several individuals intend to move to the same site, one individual is randomly chosen, and others move to the second-best site. In Figure 1(a), both individuals 2 and 3 can move to the popular site. The individual 2 was randomly chosen. This rule implements the individual’s mutual anticipation. Even a human can avoid collisions in crowded pedestrian traffic by anticipation [12]. We implement this kind of behavior by the mutual anticipation. If there is no popular site in any of the potential transition targets and another individual in the neighborhood is following, NF, defined here as a Moore neighborhood, moves due to mutual anticipation and the individual moves to occupy the absent cell generated by the mutual anticipation. Namely, it follows the predecessor (Figure 1(a), right in the center row). If an individual does not obey the mutual anticipation and is not a follower, it moves in the direction of a randomly chosen poten-

tial transition and is called a *free wanderer*. In Figure 1(a), right in the center row, the individual 3 is a free wanderer.

Figure 1(b) shows how intrinsic turbulence is generated and maintained in our swarm model. Polarization is usually used to estimate the order of coherence for a flock and a school. Polarization is defined by the length of summation for all the agents' velocities (principal vector in our model). Density is defined by the average number of agents located in the neighborhood of each agent. In Figure 1, polarization and density are normalized by maximal polarization in (a) and maximal density in (b). The inherent perturbation is defined by the number of potential transitions minus 1, normalized by the maximal number of potential transitions, which is 30 in Figure 1(b). No external noise is coupled with the transition rule for an agent.

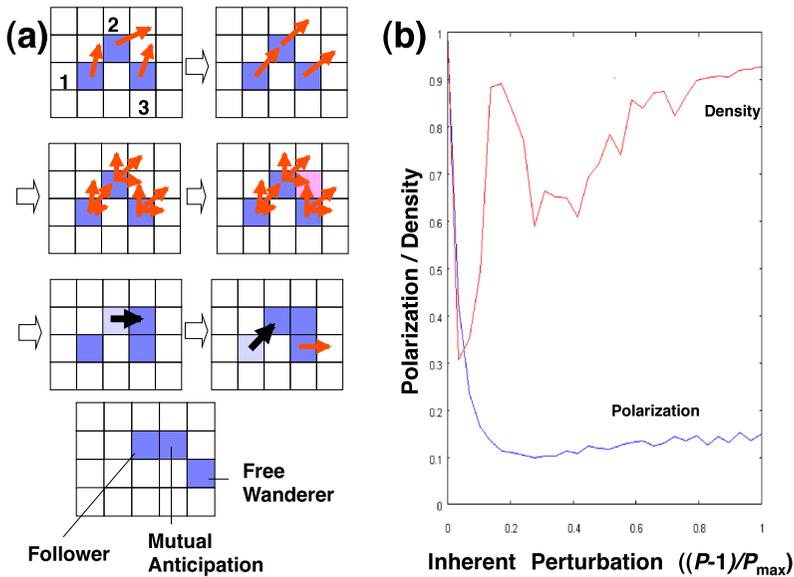


Figure 1. (a) Schematic diagram of transitions in a swarm model. Blue cells represent individuals. A pink cell represents a site with high popularity. Pale blue cells represent freshly absented cells. Each cell designates a site in a space. See text for details. (b) Polarization and density plotted against inherent perturbation. The inherent perturbation is defined by the number of potential transitions normalized by the maximum number of potential transitions.

If $P = 1$ (i.e., inherent perturbation is 0.0), mutual anticipation cannot be applied to each agent, and the model mimics the bird-like objects known as flocks [13]. Once agents are aggregated by flock centering, velocity matching makes a highly polarized flock that moves as a mass. Thus, it reveals both high polarization and high density. If $P = 2$, each agent has two potential transitions. Although it is possi-

ble that potential transitions can contribute to swarming, the overlapping of potential transition targets is too difficult to achieve mutual anticipation. The effect of multiple transitions results only in random choice of potential transitions for each agent. Thus, both density and polarization are very small. As the inherent perturbation increases, density increases and polarization decreases. High density and low polarization are maintained where inherent perturbation is larger than 0.5.

In boids and self-propelling particles (SPP) [14], a very dense flock is achieved only by high polarization. The more highly polarized a flock is, the denser it is. By contrast, low polarization breaks the coherence of a flock. The velocity matching is linearly coupled with external perturbation in boids and SPP. If the polarization is plotted against external perturbation, it decreases as the perturbation decreases. Because polarization shows a phase transition that is dependent on the external perturbation, it is regarded as an order parameter. Thus, the density of a flock is also decreased as the external perturbation decreases. In this framework, the noise disturbs the coherence of a flock, which reveals a flock that is not robust but is stable. By contrast, in our model inherent perturbation positively contributes to generating a coherent and dense swarm and can implement robust swarms, which can be used analogous to billiard balls in collision-based circuitry.

3. Collision Computing by Crab Swarm

3.1 Simulation Model

When a swarm of soldier crabs is set in a corridor, it is expected that a swarm acts as a robust ball and goes straight, and that two swarm balls are united into one ball after colliding. Because of velocity matching, a swarm ball resulting from the fusion of two balls has a summary velocity of two colliding balls.

Figure 2 shows an OR gate implemented by the collision of swarm balls. In this simulation, black regions are walls that an agent cannot invade. Agents can move in the white area. An agent moves following the rule mentioned before, where each agent has a tendency to move along the wall if it is close to the wall. This tendency is consistent with our observation of soldier crabs. In a corridor, they move along the wall. Because they have a tendency to move together, a swarm generated close to the wall of the corridor propagates along the wall. If a soldier crab is not close to the wall, it moves freely. In this OR gate an agent is assumed to move upward along the wall in Figure 2, if it is close to the wall.

When agents are set in their input position, they aggregate into one swarm moving along the wall. In this gate, a swarm initiated at either the x or y position moves upward along the central corridor after the

swarm encounters the central corridor. It is easy to see that $(x, y) = (0, 0)$ leads to $x \text{ OR } y = 0$ and that $(x, y) = (0, 1)$ and $(1, 0)$ lead to $x \text{ OR } y = 1$, where the presence of agents represents 1 and their absence represents 0. Figure 2 shows a series of snapshots of the OR gate behavior when two swarm balls are set at the x and y positions. It shows that $(x, y) = (1, 1)$ leads to $x \text{ OR } y = 1$.

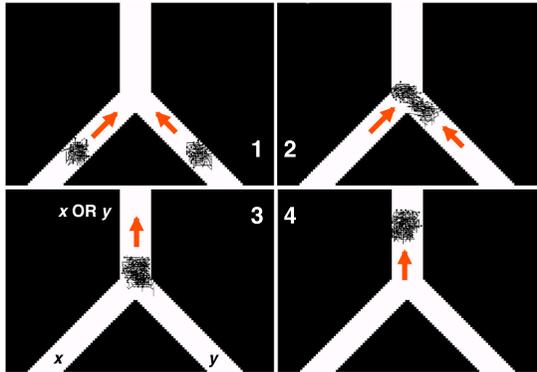


Figure 2. A series of snapshots (1, 2, 3, and 4) in an OR gate of swarm balls. Swarm balls located at the x and y positions consist of 40 agents. Each agent is represented by a square with its 5-step trajectory. Red arrows represent the direction of motion of a swarm ball.

We can estimate the effect of external perturbation if the agents' transition rule is coupled with external perturbation. As shown in Figure 2, inherent noise can positively contribute to generating and maintaining a robust swarm. Inherent noise in our model reveals multiple potential transitions, and one of them is always chosen in the transition of each agent's location. The inherent noise, therefore, cannot be distinguished from external noise. This means that external noise can contribute to generating robust swarm balls while the direction a swarm ball moves cannot be controlled. However unstable the direction of a swarm ball is, the corridor of the OR gate is one way and then the output 1 resulting from a swarm ball is robust.

Figure 3 shows an AND gate of swarm balls. Given a pair of inputs x and y located below, the left, central, and right corridors above represent the output as $\text{NOT}(x) \text{ AND } y$, $x \text{ AND } y$, and $x \text{ AND } \text{NOT}(y)$, respectively. Figure 3 shows a series of snapshots for the input $(x, y) = (1, 1)$. Two swarm balls initiated at the input locations move along their walls, leftward and rightward, respectively. Each swarm ball consists of 40 agents. After the collision, a united swarm moves upward due to the integration of their velocities and results in a united swarm moving to the central corridor, showing that $x \text{ AND } y$ is 1. As well as the behavior of an OR gate, a united swarm ball con-

sists of 80 agents and directions of transitions of agents are not analogous. A swarm ball continuously contains internal turbulence. Initial configurations of agents in these gates are randomly given in a designated area in the input position, where no external noise is coupled with the transition rule for the agent. The performance of these gates is 100 percent for OR, AND, and NOT gates. Because we can implement OR, AND, and NOT gates, we can calculate any propositional logic sentence.

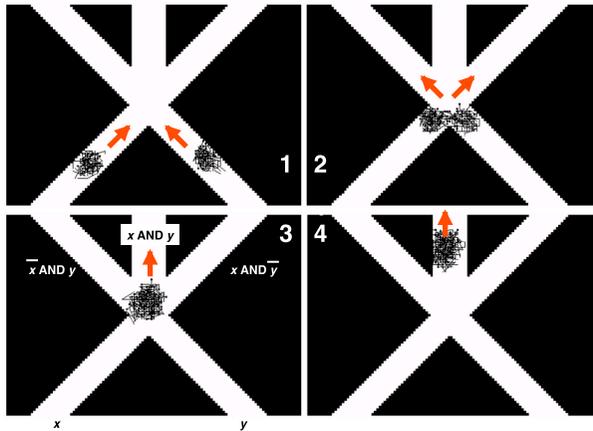


Figure 3. A series of snapshots (1, 2, 3, and 4) in an AND gate of swarm balls. Swarm balls located at the x and y positions consist of 40 agents. Each agent is represented by a square with its 5-step trajectory. Red arrows represent the direction a swarm ball is moving toward.

Finally, we discuss the robustness of the gates. As mentioned before, the OR gate acts very well against external perturbation. Here, we estimate the performance of the AND gate under a high rate of external noise. In the simulation, the external noise that is coupled with the operation of velocity matching is defined as a random value ξ selected with equal probability from $[-\lambda, \lambda]$. When the velocity of each agent in a two-dimensional space v is projected in the x and y planes denoted by v_x and v_y , the external noise is coupled by $v_x + \xi$ and $v_y + \xi$ and then normalized to make the length of the velocity be 1. We estimate performance within the range $0.0 \leq \lambda \leq 0.2$ when comparing our model of $P = 20$ with one with $P = 1$. If $P = 20$, multiple potential transitions play a role in mutual anticipation that can contribute to collective behavior. If $P = 1$, coherence of swarm results only from velocity matching.

Figure 4 shows performance of an AND gate implemented by our swarm model under a perturbed condition. The performance is defined by the success rate of the gate. In each experiment, we set 40

agents in the input positions x and y . If 80% of the agents (64 agents) in a united swarm reach the output exit of x AND y , we determine the result for the experiment to be a success; otherwise, it is a failure. The performance is defined by the number of successful experiments divided by the total number of experiments (100 experiments). Since the swarm model with $P = 1$ corresponds to the well-known boids model, the external perturbation directly influences the collective behavior of swarms. Therefore, as the strength of perturbation increases, the performance rapidly decreases.

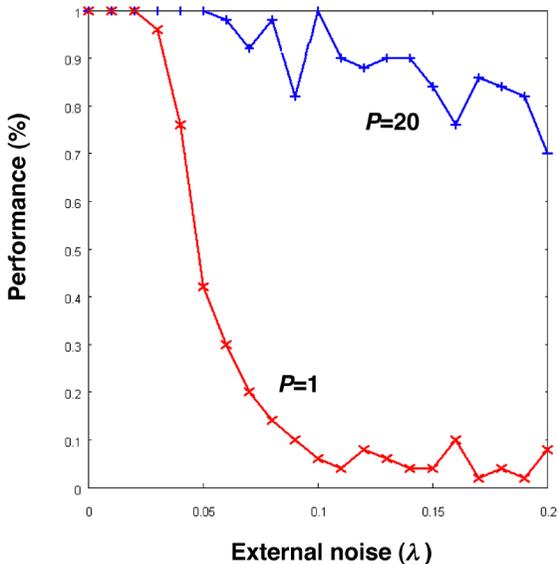


Figure 4. Performance of an AND gate implemented by the swarm model. Performance is plotted against the strength of external perturbation. A swarm model with $P = 20$ is compared to one with $P = 1$. The model with $P = 1$ can correspond to boids.

In contrast, the performance of our swarm model with $P = 20$ does not decrease drastically as the strength of perturbation increases. A large possibility of transitions can increase the probability of the presence of highly popular sites, which increases the possibility of mutual anticipation that results in the collective behavior of swarming. External perturbation also increases the possibilities of transitions. In this sense the external perturbation cannot be distinguished from potential transitions, and can produce the same contribution as increasing the number of potential transitions. The external perturbation plays a role in generating and keeping a swarm ball. The directed motion of a swarm along the wall, however, depends on the strength of velocity matching. That is the reason why the performance of the AND gate implemented by swarms with $P = 20$ is slightly decreased.

Note that the cohesive power does not weaken despite low performance.

■ 3.2 Experimental Gate Implemented by Real Soldier Crabs

We first implemented the logical gate of swarm collision by real soldier crabs, *Mictyris guinotae*. The corridor is made of acrylic plastic plates, and the gradient to move was implemented by using an orange-colored intimidation plate. The reason for having an intimidation plate is because birds are the main natural predators feeding on soldier crabs and there are usually no shadows in the lagoon except for bird shadows. This is why the crabs are very sensitive to the shadows made by standing and moving objects. Thus, the intimidation plate can trigger soldier crabs to move away from the shadowed region. The floor of the experimental gate is made of cork to provide a comfortable footing for the crabs.

Figure 5(a) shows the AND gate of swarm collision. First, we close the gate and set a swarm of soldier crabs for either or both input spaces surrounded by walls and a gate. The crabs are left for about two minutes to relax. After the crabs relax, the folded intimidation plate is extended and placed vertically; then crabs start moving along the corridor. Real-world crabs indeed behave similarly to simulated crabs. If the input for (x, y) is $(1, 0)$ or $(0, 1)$, a swarm representing the value of 1 is set by 40 crabs. If the input for (x, y) is $(1, 1)$, each swarm representing a value of 1 is set by 20 crabs.

Figure 5(b) to (e) shows the two results of the AND gate experiment for a particular swarm consisting of 40 individuals. First, 40 individuals are divided into two parts of 20 individuals each, and set for input space $(x, y) = (1, 1)$. In other cases where $(x, y) = (1, 0)$ or $(0, 1)$, 40 individuals are set for input space A or B, respectively. In any input case, both intimidation plates are unfolded and stood vertically soon after the gate is opened. The bars of Figure 5(b) to (e) show the rate (%) of individuals reaching output 1, 2, or 3. Red or blue bars represent the result of the first and second experiments for the same population of individuals. Two experiments have the same tendency, such that for the input $(1, 0)$ and $(0, 1)$, most of a swarm goes straight (i.e., A to 3 or B to 1 in Figure 5(b) to (d)), and for the input $(1, 1)$ individuals derived from input spaces A and B are united and go to the central part of the output, called 2. These results are approximated by our model simulation.

Figure 5(e) shows experimental results for 21 trials, where each trial was conducted for 40 individuals. Each bar represents the rate over all trials. The frequency histogram shows that for inputs $(1, 0)$ and $(0, 1)$, individuals of a swarm go straight or to the central corridor, meaning that most of a swarm goes straight to some extent. It also shows that for input $(1, 1)$, most of a swarm goes to the central corridor after the swarms collide. If the output of either 0 or 1 is determined by the difference in the number of individuals reaching outputs

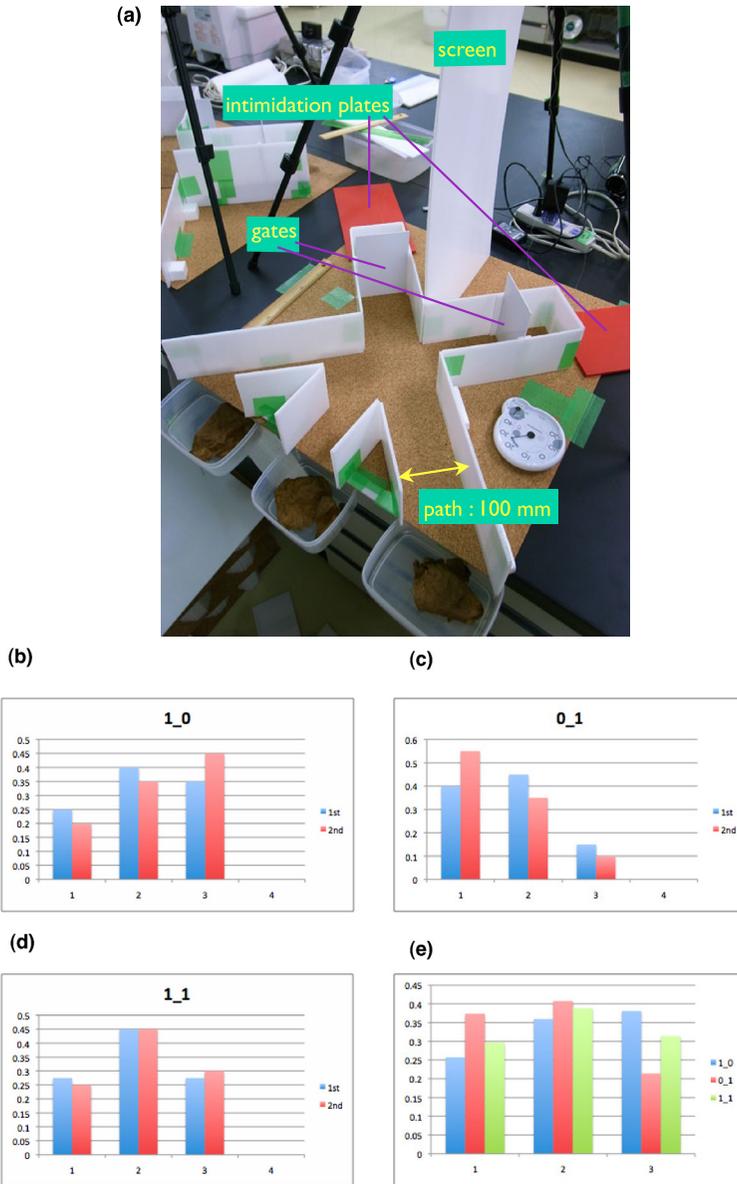


Figure 5. (a) Implementation of an AND gate for real soldier crabs. Input space for input x or y is located at the space behind the intimidation plate. The symbols 1, 2, and 3 represent output for NOT(x) AND y , x AND y , and x AND NOT(y), respectively. (b)–(e) The experimental results of AND gate implemented by real soldier crabs. (b) Frequency distribution of output 1, 2, and 3 for the input (1, 0). Output 4 represents the rate of individuals that do not reach the output in a limited time. (c) Frequency distribution for the input (0, 1). (d) Frequency distribution for the input (1, 1). (e) Frequency distribution of outputs 1, 2, and 3 over 21 trials.

1 and 3, the output is 0 if the difference exceeds 10% of all individuals (i.e., four individuals); it is 1 otherwise. If the determinant is applied to 21 trials of the AND gate experiment, performance of the AND gate for inputs (1, 0), (0, 1), or (1, 1) is 0.81, 0.76, or 0.52, respectively.

Although the experiment implemented by real soldier crabs is just a preliminary work, the performance of the AND gate is not bad. Since the OR gate can be easier to implement, the swarm collision computation can be naturally constructed. The gradient of the corridor is tested by other devices such as illumination, physical gravitational stimulus produced by slope, and some combinations. The condition of a swarm in the experiment in a room is different from that in a natural lagoon. If good enough conditions are provided for the experiment in a room, robustness of a swarm can be reconstructed.

4. Conclusion

Soldier crabs that live in tropical lagoons wander as a huge swarm in the low tidal period and live under the sand in the high tidal period. Each individual sometimes wanders freely, and is aggregated into a huge swarm triggered by a natural enemy or tidal movement dependent on the lagoon topography. Once a swarm is generated, individuals in the peripheral regions dynamically exchange their places and the internal turbulence of the crab becomes responsible for the same swarm motion. Therefore, even if the boundary of a swarm is definitely sharp and smooth, the internal structure of the swarm is dynamical and random. The mutual anticipation can negotiate perturbation and the force to make an order. In this paper, we implement ballistic computing [15] by using such a robust swarm. In the natural implementation of ballistic computation, it is difficult to refer to robustness of computation. For example, ballistic computation implemented by Belousov–Zhabotinsky (BZ) reactions is unstable without well-controlled conditions [16]. Our model suggests that biological computation is more robust even under perturbed environments.

Ethical Note

No specific license was required for this work. The duration of any single experiment was so short that no crab was ever endangered. The crabs were kept in comfortable conditions and after all experiments were released to their natural habitat. Furthermore, on visual inspection, no crabs appeared to have been injured or adversely affected by the experiments.

References

- [1] A. Adamatzky and B. De Lacy Costello, “Binary Collisions between Wave-Fragments in a Sub-Excitable Belousov–Zhabotinsky Medium,” *Chaos, Solitons & Fractals*, **34**(2), 2007 pp. 307–315. doi:10.1016/j.chaos.2006.03.095.
- [2] A. Adamatzky. “Slime Mould Logical Gates: Exploring Ballistic Approach.” (May 13, 2010) arXiv:1005.2301v1.
- [3] A. Adamatzky, B. De Lacy Costello, and T. Asai, *Reaction-Diffusion Computers*, Boston: Elsevier, 2005.
- [4] K. Yoshikawa, I. M. Motoike, T. Ichino, T. Yamaguchi, Y. Igarashi, J. Gorecki, and J. N. Gorecka, “Basic Information Processing Operations with Pulses of Excitation in a Reaction-Diffusion System,” *International Journal of Unconventional Computing*, **5**(1), 2009 pp. 3–37.
- [5] A. Toth, V. Gaspar, and K. Showalter, “Signal Transmission in Chemical Systems: Propagation of Chemical Waves through Capillary Tubes,” *Journal of Physical Chemistry B*, **98**(2), 1994 pp. 522–531. doi:10.1021/j100053a029.
- [6] A. Toth and K. Showalter, “Logical Gates in Excitable Media,” *Journal of Chemical Physics*, **103**(6), 1995 p. 2058. doi:10.1063/1.469732.
- [7] T. Nakagaki, H. Yamada, and A. Toth, “Intelligence: Maze-Solving by an Amoeboid Organism,” *Nature*, **407**, 2000 p. 470. doi:10.1038/35035159.
- [8] A. Adamatzky, *Physarum Machines: Building Computers from Slime Mould*, Singapore: World Scientific, 2010.
- [9] N. Margolus, “Physics-Like Models of Computation,” *Physica D*, **10**(1–2), 1984 pp. 81–95. doi:10.1016/0167-2789(84)90252-5.
- [10] S. Tsuda, M. Aono, and Y.-P. Gunji, “Robust and Emergent Physarum Logical-Computing,” *Biosystems*, **73**(1), 2004 pp. 45–55.
- [11] R. Rosen, *Anticipatory Systems: Philosophical, Mathematical, and Methodological Foundations*, New York: Pergamon Press, 1985.
- [12] Y.-P. Gunji, H. Murakami, T. Niizato, A. Adamatzky, Y. Nishiyama, K. Enomoto, M. Toda, T. Moriyama, T. Matsui, and K. Iizuka, “Embodied Swarming Based on Back Propagation through Time Shows Water-Crossing, Hourglass and Logic-Gate Behaviors,” in *Proceedings of the Eleventh European Conference on the Synthesis and Simulation of Living Systems (ECAL 2011)*, Paris, France (T. Lenaerts, M. Giacobini, H. Bersini, P. Bourguin, M. Dorigo, and R. Doursat, eds.), Cambridge, MA: The MIT Press, 2011 pp. 294–301.
- [13] C. W. Reynolds, “Flocks, Herds, and Schools: A Distributed Behavioral Model,” *ACM SIGGRAPH Computer Graphics*, **21**(4), 1987 pp. 25–34. doi:10.1145/37402.37406.
- [14] Y. Vicsek, A. Czirók, E. Ben-Jacob, I. Cohen, and E. Shochet, “Novel Type of Phase Transition in a System of Self-Driven Particles,” *Physical Review Letters*, **75**(6), 1995 pp. 1226–1229. doi:10.1103/PhysRevLett.75.1226.

- [15] A. Adamatzky, *Collision-Based Computing*, London: Springer, 2002.
- [16] B. De Lacy Costello, A. Adamatzky, I. Jahan, and L. Zhang, “Towards Constructing One-Bit Binary Adder in Excitable Chemical Medium,” *Chemical Physics*, **381**(1–3), 2011 pp. 88–89.
doi:10.1016/j.chemphys.2011.01.014.