

A Shortest Path Search Algorithm Using an Excitable Digital Reaction-Diffusion System

Koichi ITO^{†a)}, Masahiko HIRATSUKA^{††}, Takafumi AOKI[†], *Members*, and Tatsuo HIGUCHI^{†††}, *Fellow*

SUMMARY This paper presents a shortest path search algorithm using a model of excitable reaction-diffusion dynamics. In our previous work, we have proposed a framework of Digital Reaction-Diffusion System (DRDS)—a model of a discrete-time discrete-space reaction-diffusion system useful for nonlinear signal processing tasks. In this paper, we design a special DRDS, called an “excitable DRDS,” which emulates excitable reaction-diffusion dynamics and produces traveling waves. We also demonstrate an application of the excitable DRDS to the shortest path search problem defined on two-dimensional (2-D) space with arbitrary boundary conditions.

key words: reaction-diffusion system, nonlinear dynamics, shortest path search, excitable dynamics

1. Introduction

Living organisms can create a remarkable variety of structures to realize intelligent functions. In embryology, the development of patterns and forms is sometimes called *Morphogenesis*. In 1952, Alan Turing suggested that a system of chemical substances, called *morphogens*, reacting together and diffusing through a tissue, is adequate to account for the main phenomena of morphogenesis [1]. Recently, model-based studies of morphogenesis employing computer simulations have begun to attract much attention in mathematical biology [2], [3].

From an engineering viewpoint, the insights into morphogenesis provide important concepts for devising a new class of intelligent signal processing functions inspired by biological pattern formation phenomena [4], [5]. From this viewpoint, we have proposed a framework of *Digital Reaction-Diffusion System* (DRDS)—a discrete-time discrete-space reaction-diffusion dynamical system—for designing signal processing models exhibiting active pattern/texture formation capability. In our previous papers [6], [7], some applications of DRDS to biological texture generation and fingerprint image enhancement/restoration have already been discussed.

Manuscript received June 28, 2005.

Manuscript revised October 7, 2005.

Final manuscript received November 15, 2005.

[†]The authors are with the Department of Computer and Mathematical Sciences, Graduate School of Information Sciences, Tohoku University, Sendai-shi, 980–8579 Japan.

^{††}The author is with the Sendai National College of Technology, Sendai-shi, 989–3128 Japan.

^{†††}The author is with the Department of Electronics, Faculty of Engineering, Tohoku Institute of Technology, Sendai-shi, 982–8577 Japan.

a) E-mail: ito@aoki.ecei.tohoku.ac.jp

DOI: 10.1093/ietfec/e89–a.3.735

The DRDS can simulate a variety of reaction-diffusion dynamics by changing its nonlinear reaction kinetics. This paper describes the design of an *excitable* DRDS based on FitzHugh-Nagumo-type dynamics [2]; the designed DRDS creates excitable traveling waves exhibiting the following characteristics: (i) the waves propagate with a constant velocity, and (ii) they vanish in collisions with other waves without any other interaction. The goal of this paper is to propose an algorithm for shortest path search in two-dimensional (2-D) space using the excitable DRDS. We first define a 2-D map (specifying collision-free space and blocked space) as a boundary condition for the excitable DRDS, and initiate a traveling wave at the starting point in the map. The traveling wave propagates through the map splitting into different groups of wavefronts at branch points. A snapshot of wavefronts represents an equidistant surface measured from the starting point. By tracing back the equidistant surfaces of different time steps, we can find the shortest path from the starting point to any specified destinations.

So far, there are some papers discussing the mechanism of finding the collision-free shortest path in a 2-D map using excitable reaction-diffusion dynamics. In the papers [8]–[13], the real chemical reaction, called Belousov-Zhabotinsky (BZ) reaction, is employed as an excitable medium to generate traveling waves for path finding. The use of real chemical media for performing practical computing tasks has the weakness of limited stability in its operation. Also, the size and complexity of maps that can be handled in chemical computers may be limited. The other related papers basically employ continuous-time models of excitable dynamics, including a PDE (Partial Differential Equation) model [14] and circuit models [11], [15]. All these works focus on the mechanism of generating equidistant surfaces for the given map by using excitable chemical waves and describe only simple examples of small maps. On the other hand, this paper describes a concrete algorithm for shortest path search (including the process of tracing back the equidistant surfaces). The proposed algorithm is based on the discrete-time discrete-space model of DRDS, which is easily implemented in digital computers, and can be applied to arbitrary maps of practical size and complexity.

This paper is organized as follows: Sect. 2 defines an excitable DRDS, and presents some examples of wave propagation in the excitable DRDS. Section 3 describes a shortest path search algorithm using the excitable DRDS. Section 4 demonstrates some experiments for the shortest path

finding. Section 5 ends with some concluding remarks.

2. Excitable Digital Reaction-Diffusion System

A Digital Reaction-Diffusion System (DRDS)—a model of a discrete-time discrete-space reaction-diffusion dynamical system—can be naturally derived from the original reaction-diffusion system defined in continuous space and time (see [6] for basic mathematical discussions). The general M -morphogen DRDS can be obtained as

$$\begin{aligned} \mathbf{x}(n_0+1, n_1, n_2) \\ = \mathbf{x}(n_0, n_1, n_2) + \mathbf{R}(\mathbf{x}(n_0, n_1, n_2)) \\ + \mathbf{D}(l * \mathbf{x})(n_0, n_1, n_2), \end{aligned} \quad (1)$$

where n_0 is a time index, n_1 and n_2 are spatial indices, and

$$\mathbf{x} = [x_1, x_2, \dots, x_M]^T,$$

x_i : concentration of the i -th morphogen,

$$\mathbf{R} = T_0 \tilde{\mathbf{R}} = [R_1(\mathbf{x}), R_2(\mathbf{x}), \dots, R_M(\mathbf{x})]^T,$$

$R_i(\mathbf{x})$: nonlinear reaction kinetics for the i -th morphogen,

T_0 : time sampling interval,

$$\mathbf{D} = \text{diag}[D_1, D_2, \dots, D_M],$$

diag : diagonal matrix,

D_i : diffusion coefficient of the i -th morphogen,

$l(n_1, n_2)$

$$= \begin{cases} \frac{1}{T_1^2} & (n_1, n_2) = (-1, 0), (1, 0) \\ \frac{1}{T_2^2} & (n_1, n_2) = (0, -1), (0, 1) \\ -2(\frac{1}{T_1^2} + \frac{1}{T_2^2}) & (n_1, n_2) = (0, 0) \\ 0 & \text{otherwise,} \end{cases}$$

T_1, T_2 : space sampling intervals.

The operator $*$ in Eq. (1) denotes the spatial convolution defined as

$$\begin{aligned} (l * \mathbf{x})(n_0, n_1, n_2) \\ = \begin{bmatrix} (l * x_1)(n_0, n_1, n_2) \\ (l * x_2)(n_0, n_1, n_2) \\ \vdots \\ (l * x_M)(n_0, n_1, n_2) \end{bmatrix} \\ = \begin{bmatrix} \sum_{p_1=-1}^1 \sum_{p_2=-1}^1 l(p_1, p_2) x_1(n_0, n_1 - p_1, n_2 - p_2) \\ \sum_{p_1=-1}^1 \sum_{p_2=-1}^1 l(p_1, p_2) x_2(n_0, n_1 - p_1, n_2 - p_2) \\ \vdots \\ \sum_{p_1=-1}^1 \sum_{p_2=-1}^1 l(p_1, p_2) x_M(n_0, n_1 - p_1, n_2 - p_2) \end{bmatrix}. \end{aligned}$$

The DRDS described by Eq. (1) can be understood as a

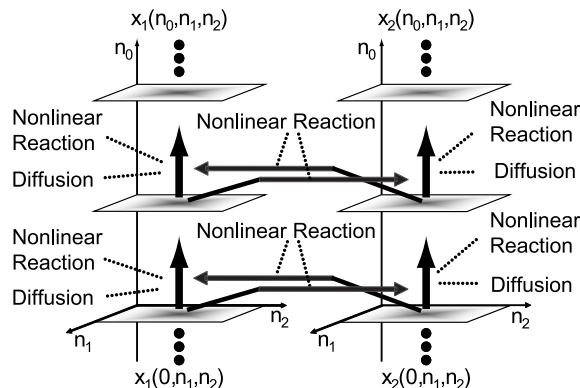


Fig. 1 Computational flow of DRDS with two morphogens ($M = 2$).

three-dimensional nonlinear digital filter. Figure 1 illustrates the computational flow of DRDS with two morphogens ($M = 2$). We first store an initial (input) image in $x_1(0, n_1, n_2)$ at time 0. After computing the dynamics for n_0 steps, we can obtain the output image from $x_1(n_0, n_1, n_2)$ at time n_0 .

DRDS can simulate various reaction-diffusion dynamics by changing the nonlinear reaction kinetics \mathbf{R} and its parameters. In this paper, we use the FitzHugh-Nagumo (FHN) model, which is one of the most widely studied excitable models [2]. The two-morphogen FHN-based DRDS, called the *excitable DRDS*, is defined as follows:

$$\begin{aligned} \begin{bmatrix} x_1(n_0+1, n_1, n_2) \\ x_2(n_0+1, n_1, n_2) \end{bmatrix} &= \begin{bmatrix} x_1(n_0, n_1, n_2) \\ x_2(n_0, n_1, n_2) \end{bmatrix} \\ &+ \begin{bmatrix} R_1(x_1(n_0, n_1, n_2), x_2(n_0, n_1, n_2)) \\ R_2(x_1(n_0, n_1, n_2), x_2(n_0, n_1, n_2)) \end{bmatrix} \\ &+ \begin{bmatrix} D_1(l * x_1)(n_0, n_1, n_2) \\ D_2(l * x_2)(n_0, n_1, n_2) \end{bmatrix}, \end{aligned} \quad (2)$$

where

$$\begin{aligned} R_1(x_1, x_2) &= T_0 \left\{ \frac{1}{k_1} \{x_1 \{x_1 - k_2\} \{1 - x_1\} - x_2\} \right\}, \\ R_2(x_1, x_2) &= T_0 \{x_1 - k_3 x_2\}. \end{aligned}$$

In this paper, we employ the parameter set: $k_1 = 10^{-3}$, $k_2 = 10^{-6}$, $k_3 = 0.1$, $D_1 = 2$, $D_2 = 0$, $T_0 = 10^{-3}$, and $T_1 = T_2 = 1$.

The excitable DRDS exhibits the characteristic behavior of excitable dynamics and generates traveling waves depending on the initial condition. Assume that the initial condition is given by $x_1(0, n_1, n_2) = x_2(0, n_1, n_2) = 0$ except for the starting point (n_1^S, n_2^S) . When we give a stimulus above the threshold (~ 0.9 for the above parameter set) at the starting point, for example, $x_1(0, n_1^S, n_2^S) = 0.9$, a traveling wave is initiated from the starting point and propagates with a constant velocity as the time step n_0 increases.

Consider an excitable DRDS of size 128×128 , where n_1 and n_2 are defined as $0 \leq n_1 \leq 127$ and $0 \leq n_2 \leq 127$. Figure 2 shows the wave propagation observed in the snapshots of the first morphogen $x_1(n_0, n_1, n_2)$. In this example, we first give initial stimuli as $x_1(0, 32, 64) = x_1(0, 96, 64) =$

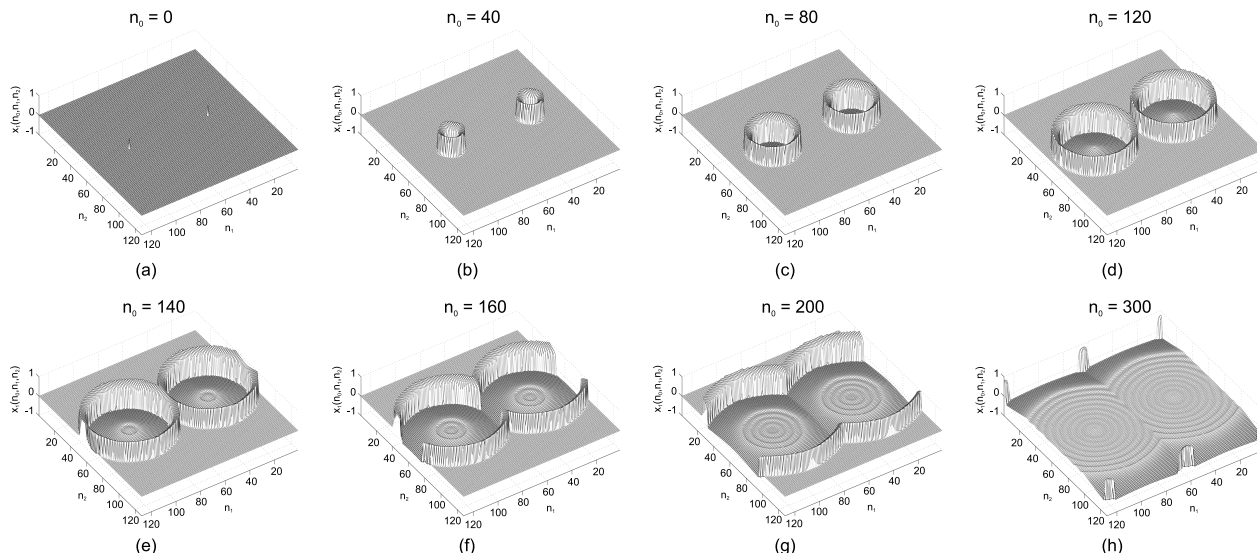


Fig. 2 Wave propagation in the 2-D excitable DRDS: (a) initial condition, (b)–(h) snapshots of wave propagation.

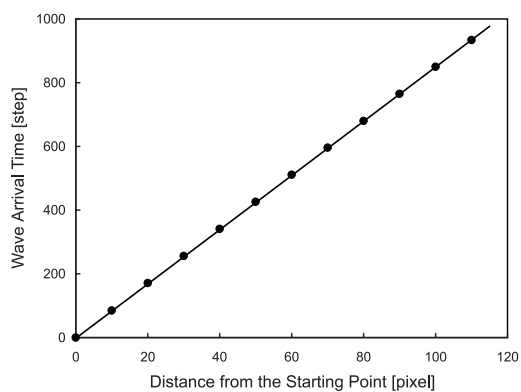


Fig. 3 Time step n_0 when the traveling wave starting at the point $(0, 0)$ arrives at the point $(n_1, 0)$.

0.9 (Fig. 2(a)). The traveling waves spread in a circular pattern and vanish in collisions with another wave as shown in Figs. 2(b)–(h).

In the above example, we can observe two important characteristics of excitable waves: (i) the waves propagate with a constant velocity and (ii) they vanish in collisions with other waves. As for the item (i) above, Fig. 3 shows the arrival time (in time step n_0) of a traveling wave as a function of the distance (in pixels) from the starting point. Note that the arrival time of a traveling wave is defined as the time when the concentration x_1 exceeds the threshold value 0.9 for the first time. We can observe that the traveling wave propagates with a constant velocity of 0.11 pixels per a unit time step. These features suggest a unique algorithm for the shortest path search problem as described in Ref. [8], where snapshots of real propagating chemical waves are considered as a collection of equidistant surfaces and are useful for finding the shortest path from the starting point to any specified points in 2-D space.

3. Shortest Path Search Algorithm

The original experiment of the shortest path search using actual chemical waves is found in Ref. [8], where the optimal pathways were determined by the collection of time-lapse position information on actual chemical waves propagating through 2-D mazes prepared with the Belousov-Zhabotinsky (BZ) reaction. Inspired by the natural computing using chemical wave propagation, we propose a shortest path search algorithm using the excitable DRDS designed in the above section. The proposed algorithm employs the excitable DRDS for wavefront generation and performs the traceback of traveling wavefronts to find the shortest paths.

Figures 4 (a)–(c) show the wave propagation in a 2-D excitable DRDS of the size 128×128 pixels. Note that we employ the boundary condition defined by obstacles in the map, where we set fixed concentrations $x_1 = 0$ and $x_2 = 0$ for obstacle locations. In Fig. 4(d), 29 snapshots of wavefronts of the first morphogen x_1 are superimposed at every 100-step intervals to form a composite image. Each wavefront represents a set of equidistant locations measured from the starting point, and hence we can derive the shortest path by tracing back the history of wavefront position from the goal to the starting point.

The proposed algorithm consists of two operations: *Forward Operation* and *Backward Operation*. Forward Operation is to generate a traveling wave in the excitable DRDS and record snapshots of equidistant wavefronts with specific time intervals. Backward Operation, on the other hand, is to trace back the wavefronts from the goal to the starting point to find the shortest pathways. We can obtain the shortest path by connecting every pair of two points on the adjacent equidistant wavefronts with the shortest distance. Figures 5 and 6 show the detailed algorithms for Forward and Back-

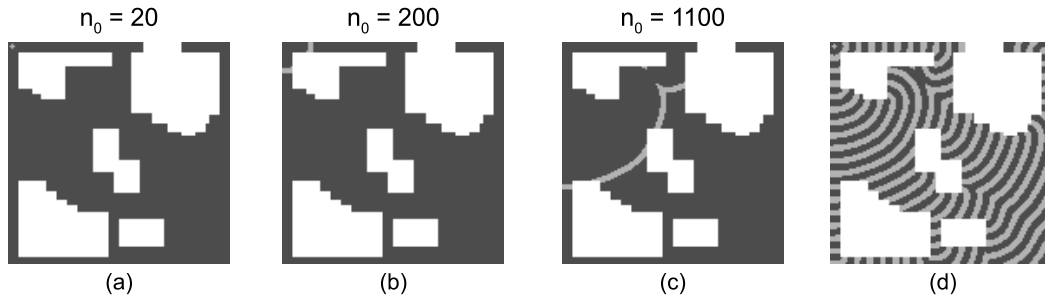


Fig. 4 Wave propagation in the 2-D excitable DRDS with obstacles appearing as white parts: (a)–(c) the snapshots of wave propagation in $x_1(n_0, n_1, n_2)$, (d) superposition of traveling waves in $x_1(n_0, n_1, n_2)$ taken every 100 steps.

procedure *Forward Operation*

Input

a starting point (n_1^S, n_2^S) , a goal (n_1^G, n_2^G) , a map (with obstacle information);

Output

$W(n_0)$: a list of points (n_1, n_2) in 2-D space at which the value of $x_1(n_0, n_1, n_2)$ is higher than a specific threshold value 0.9 (that is, $W(n_0)$ stores the list of points at which the traveling wave exists),

n_0^G : the time step when the traveling wave arrives at the goal (n_1^G, n_2^G) ;

1. **begin**

2. $n_0 \leftarrow 0$; { Initialize the time step }

3. $x_1(0, n_1, n_2) \leftarrow 0$ for all the points (n_1, n_2) ;

4. $x_2(0, n_1, n_2) \leftarrow 0$ for all the points (n_1, n_2) ;

5. $x_1(0, n_1^S, n_2^S) \leftarrow$ a constant (> 0.9);

6. $W(0) \leftarrow \{(n_1^S, n_2^S)\}$;

7. **repeat**

8. Compute the excitable DRDS (Eq. (2)) for one step assuming the boundary condition defined by the map, and derive $x_1(n_0 + 1, n_1, n_2)$ and $x_2(n_0 + 1, n_1, n_2)$;

9. Store the points of the wavefronts into $W(n_0 + 1)$ (that is, the points at which the value of $x_1(n_0 + 1, n_1, n_2)$ is higher than the threshold value 0.9);

10. $n_0 \leftarrow n_0 + 1$

11. **until** the traveling wave arrives at (n_1^G, n_2^G) ;

12. $n_0^G \leftarrow n_0$

13. **end.**

Fig. 5 Algorithm for *Forward Operation*.

ward Operations, respectively.

3.1 Forward Operation

We illustrate here the detailed algorithm for Forward Operation. The inputs of Forward Operation are a starting point (n_1^S, n_2^S) , a goal (i.e., a destination point) (n_1^G, n_2^G) and a map with obstacle information. The map defines the obstacle positions as the boundary condition of zero morphogen concentrations ($x_1 = x_2 = 0$) to block the wave propagation into obstacle regions. The outputs of Forward Operation are $W(n_0)$ and n_0^G , where $W(n_0)$ is a list of pixel positions (n_1, n_2) at which the value of the first morphogen $x_1(n_0, n_1, n_2)$ is higher than a threshold value 0.9. In other words, $W(n_0)$ stores, for every time step n_0 , the list of positions at which the traveling wave exists. The other output n_0^G is the time step when the traveling wave arrives at the goal (n_1^G, n_2^G) . The total process of Forward Operation consists of two sections: (i) initial setting and (ii) computation of excitable DRDS.

(i) Initial Setting (lines 2–6)

First, the time step n_0 is initialized to 0. For all the positions (n_1, n_2) except for the starting point (n_1^S, n_2^S) , the initial condition is given by $x_1(0, n_1, n_2) = x_2(0, n_1, n_2) = 0$. At the starting point (n_1^S, n_2^S) , the initial condition is given by $x_1(0, n_1^S, n_2^S) > 0.9$ (0.9: threshold) and $x_2(0, n_1^S, n_2^S) = 0$, which initiates a traveling wave from the starting point. $W(0)$ stores a list of pixel positions (n_1, n_2) at which the initial value of the first morphogen $x_1(0, n_1, n_2)$ is higher than the threshold value 0.9. Hence, $W(0)$ stores only the starting position (n_1^S, n_2^S) at first.

(ii) Computation of Excitable DRDS (lines 7–12)

In this step, we compute the excitable DRDS (Eq. (2)) for one step assuming the boundary condition defined by the map and derive the updated values $x_1(n_0 + 1, n_1, n_2)$ and $x_2(n_0 + 1, n_1, n_2)$. Also, the pixel positions (n_1, n_2) of the wavefronts at which $x_1(n_0 + 1, n_1, n_2) > 0.9$ are stored into $W(n_0 + 1)$. Until the traveling wave arrives at the goal (n_1^G, n_2^G) , we repeat the same task by incrementing n_0 . When the traveling wave arrives at the goal (n_1^G, n_2^G) , the time step

```

procedure Backward Operation
  Input
     $(n_1^S, n_2^S), (n_1^G, n_2^G), W(n_0), n_0^G,$ 
     $\Delta$ : a resolution of the time step interval for Backward Operation ( $\Delta = 4$  in our experiments);
  Output
    Paths: a list of points (with their time index) on the shortest path from  $(n_0^G, n_1^G, n_2^G)$  to  $(0, n_1^S, n_2^S)$ ;

  1. begin
  2.   Search  $\leftarrow \langle (n_0^G, n_1^G, n_2^G) \rangle$ ;
  3.   Paths  $\leftarrow$  Backtrace(Search)
  4. end.

function Backtrace(Search)
  1. begin
  2.    $(n_0^b, n_1^b, n_2^b) \leftarrow$  the last element of Search;
  3.   if  $n_0^b - \Delta \leq 0$  then
  4.     Search  $\leftarrow$  append{Search,  $\langle (0, n_1^S, n_2^S) \rangle$ };
  5.     return Search
  6.   else
  7.     begin
  8.       C  $\leftarrow$  get the points in  $W(n_0^b - \Delta)$  that have the shortest distance from  $(n_1^b, n_2^b)$ ;
  9.       C  $\leftarrow$  BranchDetection(C);
  10.      Search  $\leftarrow \bigcup_{(n_1^c, n_2^c) \in C} \text{Backtrace}(\text{append}\{\textit{Search}, \langle (n_0^b - \Delta, n_1^c, n_2^c) \rangle\})$ ;
  11.      return Search
  12.    end
  13. end.

function BranchDetection(C)
  1. begin
  2.   C'  $\leftarrow$  get the points in  $W(n_0^b - 2\Delta)$  that have the shortest distance from each point in C (see Fig. 7);
  3.   Calculate the distance between the two (or more) points in C';
  4.   if the calculated distance is less than  $\Delta$  then
  5.     Replace the points in C with their middle position;
  6.   return C
  7. end.

```

Fig. 6 Algorithm for Backward Operation.

is recorded in n_0^G .

3.2 Backward Operation

We describe here the detailed algorithm for Backward Operation. The inputs for Backward Operation are the starting point (n_1^S, n_2^S) , the goal (n_1^G, n_2^G) , wavefront records $W(n_0)$, the arrival time at the goal n_0^G and Δ that defines the resolution of time step interval for Backward Operation. Backward Operation uses the set of wavefront records $W(n_0^G)$, $W(n_0^G - \Delta)$, $W(n_0^G - 2\Delta)$, $W(n_0^G - 3\Delta)$, $W(n_0^G - 4\Delta)$, \dots to trace the shortest path recursively from the goal to the starting point. We use $\Delta = 4$ in our experiments. The outputs of Backward Operation is *Paths*—a list of points (with time index) on the shortest path from (n_0^G, n_1^G, n_2^G) to $(0, n_1^S, n_2^S)$. The total process of Backward Operation consists of three sections: (i) initial setting, (ii) *Backtrace*, and (iii) *BranchDetection*.

(i) Initial Setting

The search list *Search* is initialized to (n_0^G, n_1^G, n_2^G) as a possible shortest path candidate. Backtrace procedure described below is then started with the search list *Search*.

(ii) Backtrace Procedure

In this step, we find the point on the shortest path by tracing back the wavefront from the goal to the starting point. First, we get a point (n_0^b, n_1^b, n_2^b) from the last element of the search list *Search*. In the case when $n_0^b - \Delta \leq 0$, we add $(0, n_1^S, n_2^S)$ (i.e., the starting point) to the search list *Search* and return *Search*. In other cases (i.e., $n_0^b - \Delta > 0$), we load $W(n_0^b - \Delta)$, the set of wavefront points at time $n_0^b - \Delta$, and find the set of points $C (\subseteq W(n_0^b - \Delta))$ having the shortest distance from (n_1^b, n_2^b) in 2-D space. To select the points to be stored in *C*, we need to perform (iii) *BranchDetection* procedure described in the next paragraph. There is a possibility that the shortest path has multiple branches at time $n_0^b - \Delta$ and, in such a case, we store multiple points into *C*. We add each point (n_1^b, n_2^b) in *C* with its time index $n_0^b - \Delta$ to the search list *Search*, and perform *Backtrace* procedure. The *Backtrace* procedure is performed recursively until the time step $n_0^b - \Delta$ becomes less than or equal to 0 (in other words, the search point arrives at the starting point).

(iii) BranchDetection Procedure

There is a possibility that *C* (the set of wavefront points that are closest to (n_1^b, n_2^b)) contains multiple points, i.e.,

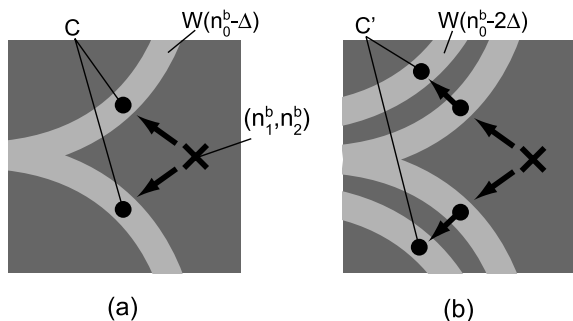


Fig. 7 Branch detection: (a) detection of multiple candidate points, and (b) branch detection using $W(n_0^b - 2\Delta)$.

$|C| \geq 2$. In such a case, we need to check whether the points in C are true branches or not. *BranchDetection* procedure further traces the wavefront backward to verify whether the set of points in C represent appropriate branches on the shortest pathways or not. For every point in C , we first find the points in $W(n_0^b - 2\Delta)$ that have the shortest distance from the point and store the detected points in C' . This process is schematically illustrated in Fig. 7. We calculate the distance between the two points in C' and classify the points in C' into groups based on the calculated distances. If the distance between the two points in C' is less than a threshold value (4 pixels in our experiments), these points are considered to be in the same group (i.e., in the same branch). We can then classify the points in C into groups according to the grouping for C' . To reduce computational complexity, we replace the points in C belonging to the same group with their middle position. The size of C directly corresponds to the number of branches at time $n_0^b - \Delta$. If there exists multiple groups in C , we can consider that the shortest path is divided into multiple pathways.

Backward Operation mentioned above can find all the possible shortest paths from the starting point to the given goal.

We can briefly confirm that the proposed algorithm can find the shortest path from the starting point (n_1^S, n_2^S) to the goal (n_1^G, n_2^G) as follows:

- Forward Operation computes the equidistant surfaces from the starting point for all the time steps n_0 ($0 \leq n_0 \leq n_0^G$). When Forward Operation terminates, the equidistant surface (at time n_0^G) arrives at the goal.
- When we start Backward Operation, clearly the shortest path from the goal to the equidistant surface at time n_0^G is obtained (since the goal is on the equidistant surface).
- After the first iteration of Backward Operation (i.e., after the first call of Backtrace function), the shortest path from the goal to the equidistant surface at time $n_0^G - \Delta$ is obtained, since the Backtrace function is designed to find the point (on the equidistant surface at time $n_0^G - \Delta$) having minimum distance from the goal.
- Similarly, after the second call of Backtrace function, the shortest path from the goal to the equidistant sur-

face at time $n_0^G - 2\Delta$ is obtained.

- This process is repeated until the Backward Operation finds the shortest path from the goal to the equidistant surface at time 0. Note that the equidistant surface at time 0 is equivalent to the starting point, and hence the shortest path from the goal to the starting point is obtained when finishing Backward Operation.

4. Experiments

This section presents some experiments of shortest path search using the proposed algorithm under three different conditions: (i) 2-D free space, (ii) map with symmetric obstacles, and (iii) maps with complicated obstacles.

In our proposed algorithm, the system parameters $k_1, k_2, k_3, D_1, D_2, T_0, T_1, T_2$ and Δ are adjusted in advance through a set of experiments. Basically, we do not need to change these parameters depending on the given problem. Our strategy of determining the adequate values of system parameters is stated as follows: (i) determine the parameters k_1, k_2, k_3 for reaction kinetics of DRDS by analyzing its phase-plane diagram so as to exhibit excitable behavior, (ii) determine the parameters D_1, D_2, T_0, T_1 and T_2 so as to generate the equidistant surfaces with a sufficient level of accuracy, and (iii) determine the time step interval Δ for Backward Operation, which controls the accuracy of the shortest path estimation for the given 2-D map. In our experiments, we employ the system parameters as $k_1 = 10^{-3}, k_2 = 10^{-6}, k_3 = 0.1, D_1 = 2, D_2 = 0, T_0 = 10^{-3}, T_1 = T_2 = 1$ and $\Delta = 4$.

(i) 2-D Free Space

Figure 8 shows the experimental result on the 2-D free space. In this example, the map is 512×512 free space, the starting point (n_1^S, n_2^S) is (3, 3), and the goal (n_1^G, n_2^G) is (509, 509). The traveling wave initiated from the starting point propagates from upper left to lower right drawing a circular pattern in Forward Operation. The shortest path from the starting point to the goal is obtained as a straight line in Backward Operation. The obtained path is obviously the shortest as shown in Fig. 8. \square

(ii) Map with Symmetric Obstacles

Figure 9 shows the experimental result on the map with symmetric obstacles. In this experiment, we obtain two shortest paths, since obstacles are symmetric to the line between the starting point and the goal. As is observed in this experiment, we can obtain two shortest paths which have equal distances. \square

(iii) Maps with Complicated Obstacles

Figure 10 shows typical examples of shortest path search in complicated mazes, where multiple goals are specified in advance. All the obtained paths avoid obstacles and are the shortest paths. \square

In the above experiments, we can observe that all the obtained paths from the starting point to the goal are shortest in terms of Euclidean distance.

We discuss here the computational complexity of the

proposed algorithm. The computation time of Forward Operation is proportional to $N_1 \times N_2 \times n_0^G$, and that of Backward Operation is proportional to n_0^G . This means that the total computation time depends directly on the size of 2-D map and the distance between the starting point and the goal. Table 1 summarizes measured computation time of the proposed algorithm for various size of 2-D maps, where the experimental results clearly show the predicted tendency. In this experiment, MATLAB ver. 6.5.1 on Pentium 4 (3.2 GHz) is employed for programming, and thus the computation time can be drastically reduced by translating the MATLAB code to other implementation-oriented

languages such as C/C++.

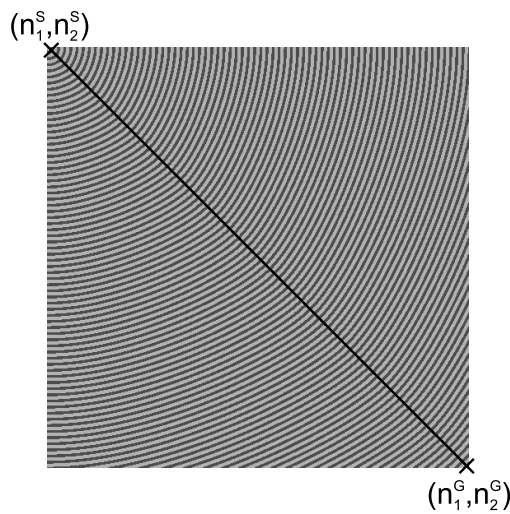


Fig. 8 Shortest path search on the 2-D free space (the wavefronts in $x_1(n_0, n_1, n_2)$ are superimposed on the map every 80 steps).

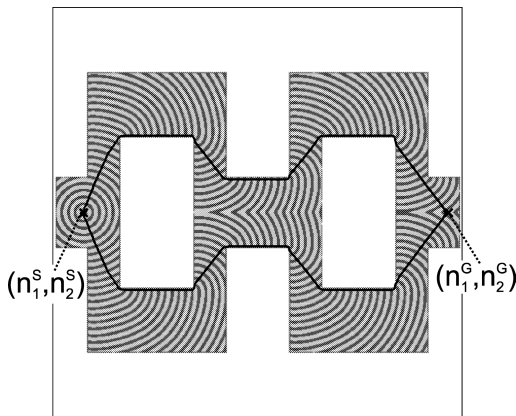


Fig. 9 Shortest path search on the map with symmetric obstacles (the wavefronts in $x_1(n_0, n_1, n_2)$ are superimposed on the map every 70 steps).

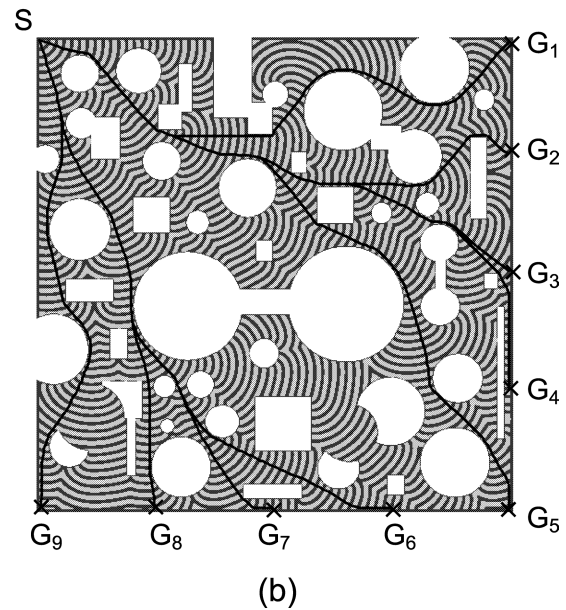
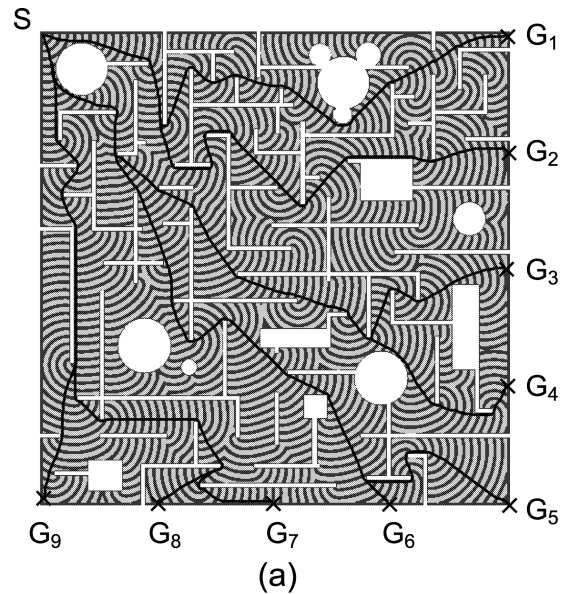


Fig. 10 Shortest path search on the maps with complicated obstacles (the wavefronts in $x_1(n_0, n_1, n_2)$ are superimposed on the map every 74 steps). S is a starting point (n_1^s, n_2^s) and G_i ($i = 1, \dots, 9$) are goals (n_1^g, n_2^g) .

Table 1 Computation time of the proposed algorithm.

$N_1 \times N_2$	(n_1^s, n_2^s)	(n_1^g, n_2^g)	n_0^G	Forward Operation	Backward Operation
64 × 64	(2, 2)	(62, 62)	1244	2.2906×10^0 sec.	1.3577×10^5 sec.
128 × 128	(3, 3)	(125, 125)	2467	3.6947×10^1 sec.	2.7203×10^0 sec.
256 × 256	(3, 3)	(253, 253)	4963	3.0242×10^2 sec.	6.6298×10^0 sec.
512 × 512	(3, 3)	(509, 509)	9980	2.2645×10^3 sec.	2.7178×10^1 sec.

5. Conclusion and Discussions

This paper presents a shortest path search algorithm in 2-D space using the excitable Digital Reaction-Diffusion System (DRDS). The traveling wave generated by the excitable DRDS has two significant features: (i) propagation with a constant velocity and (ii) annihilation in collisions with other waves. These features are effectively used to find the shortest paths in 2-D maps with various obstacles.

An alternative approach for solving large-scale shortest path problems may be to employ real chemical reactions (such as Belousov-Zhabotinsky (BZ) reaction [9], [11]) for generating traveling waves or to employ electric circuit models of excitable dynamics [16], [17]. Such approaches may have advantages of massively parallel operation for large-scale problems, and are expected to provide interesting alternatives to conventional digital computers. However, major drawbacks of using real chemical media or electric circuits are the lack of flexibility in tuning system parameters including various constants for reaction-diffusion dynamics, and the limited programmability for 2-D maps on which the shortest-path problem is solved.

On the other hand, an important advantage of the DRDS-based approach is its flexibility and programmability. We can easily modify system parameters as well as boundary conditions depending on the given problem. The proposed algorithm can be applied to various navigation tasks defined in 2-D space, and could be extended also to higher-dimensional space. A major weakness of the DRDS-based approach is its computational complexity. Clearly, this can be addressed with parallel processing technology by fully exploiting massive parallelism in DRDS computation.

The shortest path search problem discussed in this paper can be considered as an instance of discrete optimization problems—optimization problems defined over discrete structures. A subset of areas covered by discrete optimization includes transportation (vehicle routing, job scheduling, packing), industrial planning (production planning, machine scheduling, inventory management), finance (investment planning, portfolio management), molecular biology (DNA alignment), system design (VLSI design, processor scheduling, network design), etc. It is possible to define the shortest path search problem discussed in this paper in terms of discrete optimization. There are many approaches for solving discrete optimization problems including optimal methods (branch and bound, dynamic programming, etc.) and heuristics (local search, simulated annealing, genetic algorithms, etc.). Note that the DRDS-based approach is not a general-purpose framework for discrete optimization. A natural question may arise here: What class of problems could be handled by DRDS-based approach? The excitable DRDS can generate accurate equidistant surfaces from a given starting point, and hence it may be applied to some problems in computational geometry, in which the equidistant surfaces play an important role to solve the problems. Some examples of applications are Voronoi diagram

formation and thinning or skeletoning of binary images. Detailed characterization of possible applications is being left for future study.

References

- [1] A.M. Turing, "The chemical basis of morphogenesis," *Phil. Trans. Roy. Soc. London*, vol.B237, pp.37–72, Aug. 1952.
- [2] J.D. Murray, *Mathematical Biology*, Springer-Verlag, Berlin, 1993.
- [3] S. Kondo and R. Asai, "A reaction-diffusion wave on the skin of the marine angelfish pomacanthus," *Nature*, vol.376, pp.765–768, Aug. 1995.
- [4] A.S. Sherstinsky and R.W. Picard, "M-lattice: From morphogenesis to image processing," *IEEE Trans. Image Process.*, vol.5, no.7, pp.1137–1150, July 1996.
- [5] K.R. Crouse and L.O. Chua, "Methods for image processing and pattern formation in cellular neural networks: A tutorial," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol.42, no.10, pp.583–601, Oct. 1995.
- [6] K. Ito, T. Aoki, and T. Higuchi, "Digital reaction-diffusion system—A foundation of bio-inspired texture image processing," *IEICE Trans. Fundamentals*, vol.E84-A, no.8, pp.1909–1918, Aug. 2001.
- [7] K. Ito, T. Aoki, and T. Higuchi, "Fingerprint restoration using digital reaction-diffusion system and its evaluation," *IEICE Trans. Fundamentals*, vol.E86-A, no.8, pp.1916–1924, Aug. 2003.
- [8] O. Steinbock, A. Tóth, and K. Showalter, "Navigating complex labyrinths: Optimal paths from chemical waves," *Science*, vol.267, pp.868–871, Feb. 1995.
- [9] A. Adamatzky, B. De Lacy Costello, C. Melhuish, and N. Ratcliffe, "Experimental reaction-diffusion chemical processors for robot path planning," *J. Intell. Robot. Syst.*, vol.37, no.3, pp.233–249, July 2003.
- [10] A. Adamatzky and B. De Lacy Costello, "Reaction-diffusion path planning in a hybrid chemical and cellular-automaton processor," *Chaos, Solitons and Fractals*, vol.16, no.5, pp.727–736, June 2003.
- [11] A. Adamatzky, P. Arena, A. Basile, R. Carmona-Galán, B. De Lacy Costello, L. Fortuna, M. Frasca, and A. Rodríguez-Vázquez, "Reaction-diffusion navigation robot control: From chemical to VLSI analogic processors," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol.51, no.5, pp.926–938, May 2004.
- [12] A. Adamatzky, "Computing with waves in chemical media: Massively parallel reaction-diffusion processors," *IEICE Trans. Electron.*, vol.E87-C, no.11, pp.1748–1756, Nov. 2004.
- [13] N.G. Rambidi and D. Yakovenchuk, "Finding paths in a labyrinth based on reaction-diffusion media," *Adv. Mater. Opt. Electron.*, vol.9, no.1, pp.27–34, 1999.
- [14] K. Agladze, N. Magome, R. Aliev, T. Yamaguchi, and K. Yoshikawa, "Finding the optimal path with the aid of chemical wave," *Physica D*, vol.106, pp.247–254, 1997.
- [15] V. Pérez-Muñuzuri, V. Pérez-Villar, and L.O. Chua, "Autowaves for image processing on a two-dimensional CNN array of excitable nonlinear circuits: Flat and wrinkled labyrinths," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol.40, no.3, pp.174–181, March 1993.
- [16] T. Asai, Y. Nishimiya, and Y. Amemiya, "A CMOS reaction-diffusion circuit based on cellular-automaton processing emulating the Belousov-Zhabotinsky reaction," *IEICE Trans. Fundamentals*, vol.E85-A, no.9, pp.2093–2096, Sept. 2002.
- [17] T. Asai, Y. Kanazawa, T. Hirose, and Y. Amemiya, "Analog reaction-diffusion chip imitating the Belousov-Zhabotinsky reaction with hardware Oregonator model," *Int. J. Unconventional Computing*, vol.1, no.2, pp.123–147, 2005.



Koichi Ito received the B.E. degree in electronic engineering, and the M.S. and Ph.D. degree in information sciences from Tohoku University, Sendai, Japan, in 2000, 2002 and 2005, respectively. He is currently a Research Associate of the Graduate School of Information Sciences at Tohoku University. From 2004 to 2005, he was a Research Fellow of the Japan Society for the Promotion of Science. His research interest includes signal and image processing, and biometric authentication. Dr. Ito received the

Tohoku Section Presentation Award from Information Processing Society of Japan in 2000.



Masahiko Hiratsuka received the B.E. degree in electronic engineering, and the M.S. and Ph.D. degrees in information sciences from Tohoku University, Sendai, Japan, in 1995, 1997 and 2000, respectively. He is currently a Research Associate at Sendai National College of Technology, Sendai, Japan. From 1998 to 2000, he was a Research Fellow of the Japan Society for the Promotion of Science. His research interests include biomolecular computing and bioelectronics. Dr. Hiratsuka received the Outstanding Transactions Paper Award from the Institute of Electronics, Information and Communication Engineers (IEICE) of Japan in 1997, the IEICE Inose Award in 1997, and the IEE Mounbatten Premium Award in 1999.

standing Transactions Paper Award from the Institute of Electronics, Information and Communication Engineers (IEICE) of Japan in 1997, the IEICE Inose Award in 1997, and the IEE Mounbatten Premium Award in 1999.



Takafumi Aoki received the B.E., M.E., and D.E. degrees in electronic engineering from Tohoku University, Sendai, Japan, in 1988, 1990, and 1992, respectively. He is currently a Professor of the Graduate School of Information Sciences at Tohoku University. For 1997–1999, he also joined the PRESTO project, Japan Science and Technology Corporation (JST). His research interests include theoretical aspects of computation, VLSI computing structures for signal and image processing, multiple-valued logic, and

biomolecular computing. Dr. Aoki received the Outstanding Paper Award at the 1990, 2000 and 2001 IEEE International Symposiums on Multiple-Valued Logic, the Outstanding Transactions Paper Award from the Institute of Electronics, Information and Communication Engineers (IEICE) of Japan in 1989 and 1997, the IEE Ambrose Fleming Premium Award in 1994, the IEICE Inose Award in 1997, the IEE Mounbatten Premium Award in 1999, the Best Paper Award at the 1999 IEEE International Symposium on Intelligent Signal Processing and Communication Systems, and the IP Award at the 7th LSI IP Design Award in 2005.



Tatsuo Higuchi received the B.E., M.E., and D.E. degrees in electronic engineering from Tohoku University, Sendai, Japan, in 1962, 1964, and 1969, respectively. He is currently a Professor at Tohoku Institute of Technology and a Emeritus Professor at Tohoku University. From 1980 to 1993, he was a Professor in the Department of Electronic Engineering at Tohoku University. He was a Professor from 1994 to 2003, and was Dean from 1994 to 1998 in the Graduate School of Information Sciences at To-

hoku University. His general research interests include the design of 1-D and multi-D digital filters, linear time-varying system theory, fractals and chaos in digital signal processing, VLSI computing structures for signal and image processing, multiple-valued ICs, multiwave opto-electronic ICs, and biomolecular computing. Dr. Higuchi received the Outstanding Paper Awards at the 1985, 1986, 1988, 1990, 2000 and 2001 IEEE International Symposiums on Multiple-Valued Logic, the Certificate of Appreciation in 2003 and the Long Service Award in 2004 on Multiple Valued Logic, the Outstanding Transactions Paper Award from the Society of Instrument and Control Engineers (SICE) of Japan in 1984, the Technically Excellent Award from SICE in 1986, and the Outstanding Book Award from SICE in 1996, the Outstanding Transactions Paper Award from the Institute of Electronics, Information and Communication Engineers (IEICE) of Japan in 1990 and 1997, the Inose Award from IEICE in 1997, the Technically Excellent Award from the Robotics Society of Japan in 1990, the IEE Ambrose Fleming Premium Award in 1994, the Outstanding Book Award from the Japanese Society for Engineering Education in 1997, the Award for Persons of scientific and technological merits (Commendation by the minister of state for Science and Technology), the IEE Mounbatten Premium Award in 1999, the Best Paper Award at the 1999 IEEE International Symposium on Intelligent Signal Processing and Communication Systems, and the IP Award at the 7th LSI IP Design Award in 2005. He also received the IEEE Third Millennium Medal in 2000. He received the fellow grade from IEEE and SICE.