A-023

Pattern Generation on Two Dimensional Cellular Automata

渡辺 識*　　大川 知†
Satoru Watanabe　　Satoshi Okawa

1 Introduction

A dot matrix method is used to express the two dimensional patterns such as figures or characters in printers or display. In this method, as the patterns are obtained by dots on lattices of fixed size, several dot patterns for each pattern should be prepared depending on the screen size.

In this paper, we investigate the method to display patterns on the screen of any size without such preparations. First, we define two dimensional patterns as equivalence classes which are obtained by similar relation defined by movings and scaling on two dimensional plane. For an $m \times n$ screen, we define a pattern generation as to display with appropriate size (and position) in the screen. Next, we discretize the screen, and we define a pattern generation on the discretized screen. Furthermore, we study a correspondence between the discretized screen and cellular automata, and the pattern generation on the cellular automata. In the last part, we show methods for pattern generation of a square and a right isosceles triangle on two dimensional cellular automata as examples of pattern generation.

2 Pattern Generation

Let $\mathbb{R}$ be a set of real numbers, and a two dimensional plane is denoted by $\mathbb{R} \times \mathbb{R}$. A set $F \subseteq \mathbb{R} \times \mathbb{R}$ is called a two dimensional figure, a set of all two dimensional figures is denoted by $\mathcal{F}$, that is $\mathcal{F} = \{F | F \subseteq \mathbb{R} \times \mathbb{R}\}$. A figure which is obtained with moving F by $d \in \mathbb{R} \times \mathbb{R}$ is denoted by $F + d = \{p + d | p \in F\}$, and a figure which is obtained with extending F by $a (a > 0)$ times is denoted by $a \cdot F = \{a \cdot p | p \in F\}$. We define mappings S_d and Z_a as follows respectively,

$$S_d(F) = F + d, \quad Z_a(F) = a \cdot F.$$

We define a relation $\sim$ on $\mathcal{F}$ using S_d and Z_a as follows.

For $F_1, F_2 \in \mathcal{F}$,

$$F_1 \sim F_2 \Leftrightarrow F_2 = S_d Z_a(F_1) \ (= aF_1 + d).$$

The relation $\sim$ is an equivalence relation on two dimensional figures. We define a pattern as a equivalence class using this relation as follows.

Definition 1
For a figure F, a pattern $[F]$ containing F is defined by

$$[F] = \{F' | F' \sim F\},$$

and a set of pattern $\mathcal{P}$ is defined by

$$\mathcal{P} = \mathcal{F} / \sim = \{P | P = [F], F \in \mathcal{F}\}.$$

For any $m, n > 0$, $[0, m] \times [0, n] \subseteq \mathbb{R} \times \mathbb{R}$ is called a screen of size $m \times n$, and it denoted by $C_{m \times n}$, where $[a, b]$ is an interval $\{x | a \leq x \leq b\}$.

Definition 2
For a pattern $P \in \mathcal{P}$ assuming $P = [F]$, generation of P on $C_{m \times n}$ is to obtain a set $D \subseteq C_{m \times n}$ which satisfies following conditions.

1. $\exists a, d \quad D = S_d Z_a(F)$,
2. $\forall \epsilon > 0 \quad S_d Z_{a+\epsilon}(F) \not\subseteq C_{m \times n}$.

When we display a figure in a screen, the screen has to be discretized, so we discretize $C_{m \times n}$ by dividing the width by m and dividing

*The University of Aizu
†The University of Aizu

the length by n. The screen cell at the leftmost and the bottom position of the discretized screen is $c_{1,1}$. $c_{i,j}$ describes the cell which is positioned in the ith position from the left side of the array, and jth position from the bottom of the array, that is $c_{i,j} = C_{[i-1,i]\times[j-1,j]}$.

We define the screen $C_{m,n}$ which is obtained by discretizing $C_{m\times n}$ as follows,

$$C_{m,n} = \{c_{i,j} | 1 \leq i \leq m, 1 \leq j \leq m, \quad i,j \in N\}.$$

We define a pattern generation on the discretized screen as follows.

Definition 3

For a pattern $P = [F] \in \mathcal{P}$, generation of P on $C_{m,n}$ is to obtain the following set $D' \subseteq C_{m,n}$,

$$D' = \{c_{i,j} | c_{i,j} \cap D \neq \phi\}.$$

3 Implementation with Cellular Automata

Two dimensional cellular automata consist of copies of a finite automaton (cell) which are positioned such as lattices. Each cell changes its own state to the state which is determined according to its own state and the adjacent cells' states. We call the own and adjacent cells neighbors, the function to determine the next state according to neighbors' states is called a local mapping. Each cell is expressed by $a_{i,j}$, which means ith row and the jth column from the leftmost lowest cell. The interval of updating state is called a step. Formally, a two dimensional cellular automaton $\mathcal{M}$ is defined as follows,

$$\mathcal{M} = (M, Q, \sigma, N),$$

where $M \subset Z \times Z$: a coordinate set where cells exist (we assume it is connected. Z means a set of integers), Q : a set of states, $\sigma : Q \times Q^{|N|-1} \rightarrow Q$: a local mapping, N is a set of neighbors. In this paper, we investigate the automata which are placed m cells widthways and n cells lengthways, we call them $m \times n$ cellular automata, and we assume N as Neumann neighborhood, namely consisting of the own, upper, lower, right and left cells.

By regarding each cell $a_{i,j}$ as $c_{i,j}$ in the discretized screen $C_{m,n}$, the set M can be regarded as the discretized screen $C_{m,n}$, and then, an $m \times n$ cellular automaton can be denoted as follows,

$$\mathcal{M} = (C_{m,n}, Q, \sigma, N).$$

Therefore, we regard a problem to generate P on $C_{m,n}$ as a problem to generate P on a cellular automaton $\mathcal{M}$, that is, a problem to construct $\mathcal{M}$.

To construct the $\mathcal{M}$ is to provide σ which specifies $D' \subseteq C_{m,n}$ at a certain time starting from the initial configuration. D' is specified by letting $a_{i,j}$ be in a special state s if $a_{i,j} \in D'$.

Next, we explain propagation of signals among cells. When the next cell of a cell in s changes its state to s at k steps, we say the signal specified by s propagates at speed $1/k$.

For example, 1/1 means a changing a next cell's state to a specific state at 1 step, as a result, the signal which is expressed by the specific state is sent at 1 step. Similarly, 1/2 means a changing a next cell's state to a specific state at 2 steps, as a result, the signal which is expressed by the specific state is sent at 2 steps. We call these signals 1/1 signal and 1/2 signal respectively.

4 Examples of Pattern Generation on Cellular automata

As examples of pattern generation, we show the drawings of a square and a right isosceles triangle on an $m \times n$ cellular automaton as follows.

4.1 square pattern

We investigate a method to generate the square of maximum size in the center of a given $m \times n$ cellular automaton. We explain each step of the method as follows.

(1)comparison of length and width

The cell $a_{1,1}$ which is in O of Figure 1 sends 1/1 signals to $a_{2,1}$ and $a_{1,2}$ simultaneously, and then $a_{2,1}$ and $a_{1,2}$ send 1/1 signal to $a_{2,2}$. As a result, $a_{1,1}$ sends a 1/2 signal to $a_{2,2}$. Repeating this manner, the signal arrives at P on one of the edges as shown in Figure 1. In this example, as the result of comparison, $m > n$ is obtained.

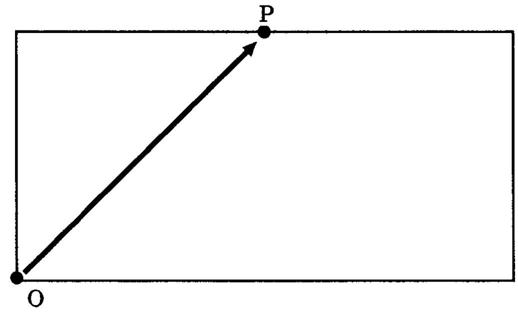

Figure 1: comparison of length and width

(2) setting of the right top corner

P sends Signal a of speed 1/1 and Signal b of speed 1/3 to right direction simultaneously as shown in Figure 2. After Signal a reached F, Signal a goes back to left direction at same speed, and then Signal a meets Signal b at C which is midpoint of P and F. The point C becomes the right upper corner of the square.

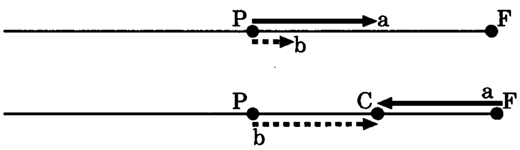

Figure 2: setting of the right top corner

(3) drawing the square

C sends Signal c of speed 1/2 to left lower direction to set the right bottom corner A. The cells that receives the Signal c sends Signal d of speed 1/1 to upper and lower direction as shown in Figure 3.

The cells that receives Signal d change their own states to the special state s, and then, we can obtain the square.

By the method mentioned above, a figure is drawn gradually. For an instantaneous appearance of the figure at a certain time, we can generate the square using the firing squad synchronization instead of the method after step (3). The

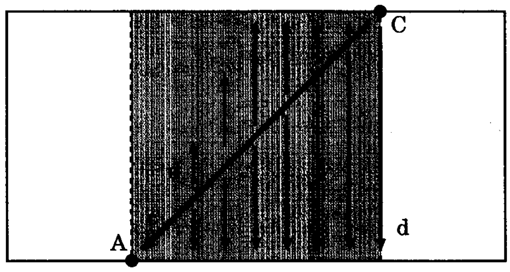

Figure 3: drawing the square

following method shows a generation by the firing squad synchronization technique.

(3′) setting of the corners

C sends Signal c of speed 1/1 to the lower direction and Signal d of speed 1/2 to the left lower direction simultaneously, and then Signal c and d reach B and A which become the bottom corners of the square respectively as shown in Figure 4.

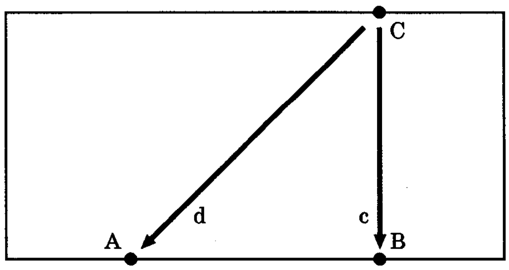

Figure 4: setting of the bottom corners

After Signal d reaches A, A sends Signal e of speed 1/1 to upper direction, and then Signal e reaches D which becomes left top corner of the square as shown in Figure 5.

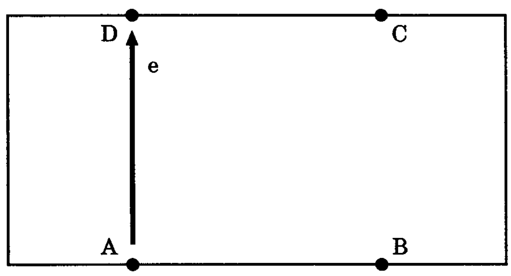

Figure 5: setting of the left top corner

(4′) drawing the square by the firing squad synchronization technique

After corners are obtained, cells draw the square by the firing squad synchronization technique to the internal of ABCD, and then we can obtain the square as shown in Figure 6.

For shortening the time of drawing, starting from B which is the general, the drawing can be begun at time that B is obtained. The signals for the firing squad synchronization are sent before the corners A and D are obtained, but the corners are obtained before the signal from B reaches them.

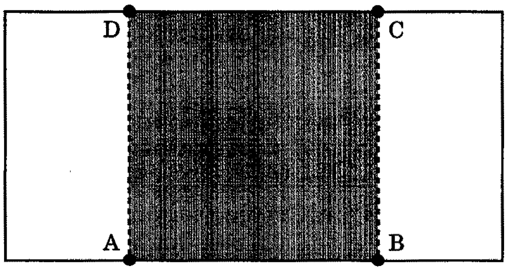

Figure 6: drawing the square by firing squad synchronization

4.2 Right isosceles triangle pattern

We investigate a method to generate the right isosceles triangle of maximum size that base is equal to the right side in the center of a given $m \times n$ cellular automaton. We explain each step of the method as follows.

(1) setting of the top vertex

We can obtain the top vertex C by the same manner of drawing the square.

(2) drawing the right isosceles triangle form

C sends Signal c of speed 1/2 to left lower direction to set the right bottom corner A. The cells that receives the Signal c sends Signal d of speed 1/1 to lower direction as shown in Figure 7. The cells that receives Signal d change their own states to the special state s, and then we can obtain the right isosceles triangle form.

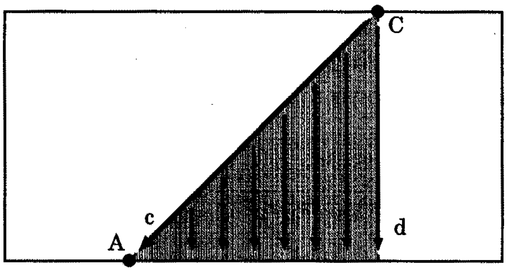

Figure 7: drawing the right isosceles triangle form

5 Conclusion

In this paper, we defined a pattern and a pattern generation. Next, we discretized the screen, and defined a pattern generation on the discretized screen. Furthermore, we studied a correspondence between the discretized screen and cellular automata, and we studied the pattern generation on the cellular automata. In the last part, we explained methods for pattern generation of the square and the right isosceles triangle on a two dimensional cellular automaton as examples of pattern generation.

We have studied to generate more complex patters or characters which consist of curves or combinations of curves and straight lines. At present, we have to consider a method for pattern generation according to an individual figure. If any pattern can be generated using a standardized method, we can use it in a display such as an electric bulletin board or a printer.

reference

[1] M.Teraoka, et al. A Design of Generalized Optimum-Time Firing Squad Synchronization Algorithm for Two-Dimensional Cellular Arrays, *The 18th Annual Conf.of Japanese Society for Artificial Intelligence*, 3H1-04,2004.

[2] T.Komatsu, Pattern Generation in Two Dimension Plane, *University of Aizu, Grad.Thesis.*, March, 2008.

[3] S.Wolfram, Two-Dimensional Cellular Automata, *Journal of Statistical Physics, vol 38, p901-946*, March 1985.