Regular Triangle Generation on Two-Dimensional Cellular Automata

渡辺 識   大川 知

Satoru Watanabe   Satoshi Okawa

会津大学

The University of Aizu

1 Introduction

In a dot matrix method, as a pattern or a figure are obtained by dots on lattices of fixed size, several dot patterns for each pattern should be prepared depending on the screen size. In recent years, as a method which is independent from screen size, several methods to draw some patterns on a screen which consists of two dimensional cellular automata have been proposed[4][6][9][10]. Komatsu has proposed methods to draw a square and a rectangle of maximum size in the center of the screen. Watanabe and Okawa have proposed another method to draw a square and a diamond[6], and a method to draw a circle of radius r at the center of the screen[9][10] .

In this paper, we review the definitions of pattern and pattern generation, and we investigate an algorithm to draw a regular triangle of maximum size at the center of the screen. First, we define two dimensional patterns as equivalence classes which are obtained by the similarity relation defined by movings and scaling on two dimensional plane. For an $m \times n$ screen, we define a pattern generation to display with appropriate size (and position) in the screen, and we define a pattern generation on the discretized screen. Next, we study a correspondence between the discretized screen and cellular automata, and we study the pattern generation on the cellular automata. Furthermore, we review a basic signal propagation and a method to obtain a mid point of one dimentional cellular array, and we investigate a method to draw a line segment between any two points and a line through a given point and parallel to a given line as fundamental techniques for geometorical drawings. In the last part, we show an algorithm to draw a regular triangle of maximum size at the center of the screen.

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

Following discussion, we assume that m and n are integers for simplicity. When we display a figure on a screen, the screen has to be discretized, so we discretize $C_{m \times n}$ by dividing the width by $m - 1$ and dividing the length by $n - 1$. In this process, for each lattice point p, a copy of small screen is set on it. The small screen at the leftmost and the bottom position of the discretized screen is $c_{0,0}$, and a screen which is positioned in the ith position from the left side of the array and jth position from the bottom of the array is described by $c_{i,j}$, that is

$c_{i,j} = C_{[i-0.5, i+0.5] \times [j-0.5, j+0.5]}$.

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

$$C_{m,n} = \{c_{i,j}|0 \leq i \leq m, 0 \leq j \leq n, \quad i, j \in N\}.$$

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 $M \subset Z \times Z$ for an integer set Z : a connected coordinate set to express positions of cells,

Q: a set of states, σ: a local mapping from $Q \times Q^{|N|-1}$ to Q, N: 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. In an initial configuration of $\mathcal{M}$, $a_{0,0}$ is active, and the all other cells are in *quiescent*.

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, for pattern P, 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 $\mathcal{M}$ which generates P.

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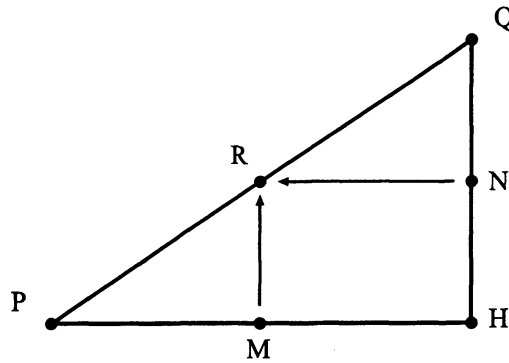

Figure 1: Generation of A Line Segment

4 Fundamental Techniques for Geometorical Drawings

We explain some techniques for geometorical drawings which are used in a generation of a regular triangle in two dimensional cellular automata.

4.1 Basic Signal Propagation

When a next cell of a cell in state s changes its own state to s at k steps, we call the signal specified by s propagates at speed $1/k$. A cell can send signals upper, lower, right, and left directions.

We can obtain a cell at the mid point of one dimentional cellular array by using defferent speed of signals. A cell A which is an end point of the array sends a signal s with speed $1/1$ and a signal t with speed $1/3$ to the right direction simultaneously. A cell B which is another end point of the array receives the signal s, and then the cell B sends a signal $\bar{s}$ with speed $1/1$ to the left direction. The signal $\bar{s}$ and the signal t meet at a cell M, the mid point of the array.

4.2 Generation of A Line Segment

For any cells P and Q, we explain how to draw a line segment between them in Figure 1 as follows. P sends a signal right direction and Q sends lower direction respectively, and then the two signals hit each other in a cell H. The cell H sends signals left and upper direction to

obtain the mid points M and N of PH and QH respectively. The cell M sends a signal upper direction, and the cell N sends a signal left direction, and then the two signals hit each other in cell R which is the mid point of the line segment PQ. Repeating the above method recursively, we obtain the line segment PQ.

4.3 Genaration of A Parallel Line

For any cell P and a line l, we explain how to draw a line through P and parallel to l as shown in Figure 2. The cell P sends a signal s upper direction and sends a signal t right direction. The signal s hits a cell P' on the line l, and then P' sends a signal s' right direction. The signal s' hits a certain cell Q which is put beforehand, and then Q sends a signal u upper direction, and a signal v lower direction simultaneously.
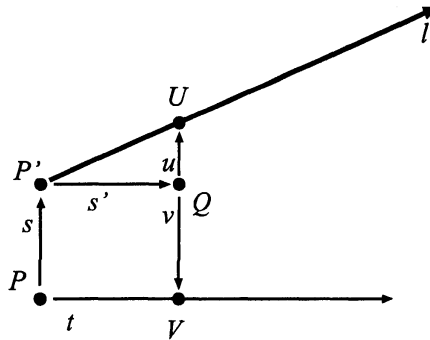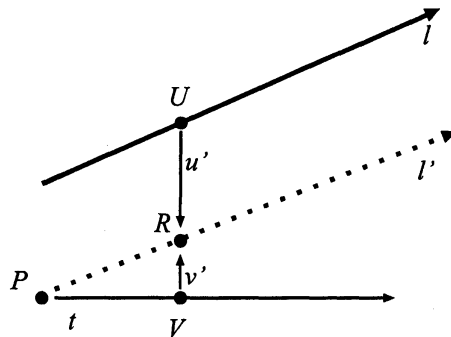

Figure 2: Generation of A Parallel Line

Figure 3: Generation of A Parallel Line

The signal u hits a cell U on the line l and the cell U returns a signal u' lower directon. The signal v hits a cell V on the trace of signal t, and the cell V returns a signal v' upper directon. The signal u' and the signal v' meet at a cell R each other as shown in Figure 3. Drawing a line between the cells P and R by the method explained in Sec.4.2, we obtain a prallel line l'.

5 Regular Triangle Generation on Cellular automata

We investigate a method to generate a regular triangle of maximum size at the center of a given $m \times n$ cellular automaton. In the following example, we assume that for $m > n$, the maximum square A, B, C and D at the center of the screen, the arc, BD, quarter of a circle with radius AB and center A are obtained beforehand [6] as shown in Figure 4.
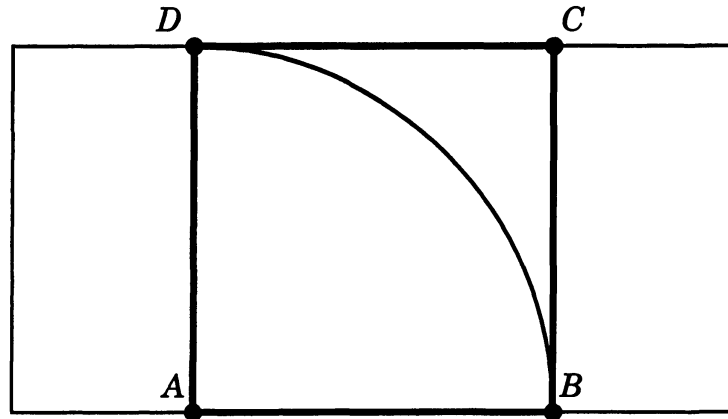
Figure 4: setting the square and an arc on the screen

5.1 A Method to Generate a Regular Triangl

We explain the generation of a regular triangle at the center of the screen step by step.

(1) Top Vertices of The Regular Triangles

The cell A sends signals to right direction and we obtain cell M which is a mid point of AB. The mid point cell M of AB can be obtained by the method explained in Sec.4.1. Then, a signal which is sent upward from M passes the cell T on arc BD and reaches cell T' on line CD. Here, T is a top vertex of a regular triangle ABT and T' is the top vertex of the regular triangle we will draw.
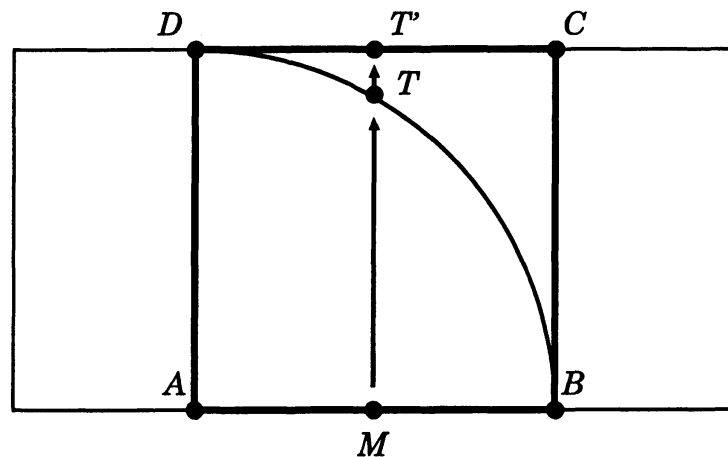

Figure 5: Top Vertex of The Regular Triangle

(2) Sides of The Regular Triangle

For A, B, and T, line segments AT and BT can be obtained by the method explained Sec.4.2 and a regular triangle ABT is obtained.

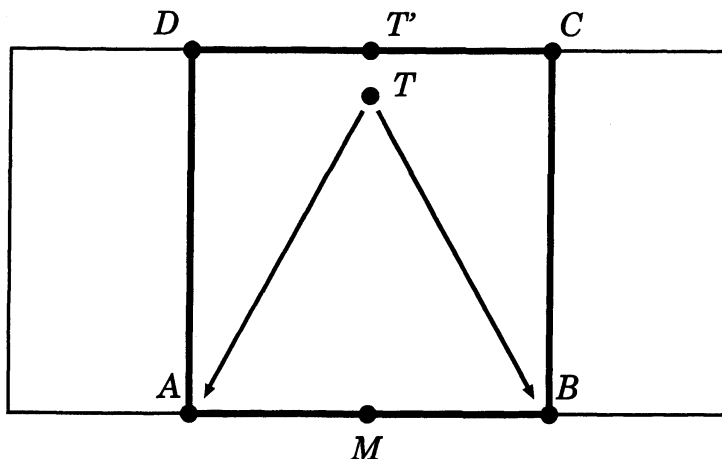### (3) Expansion of The Regular Triangle
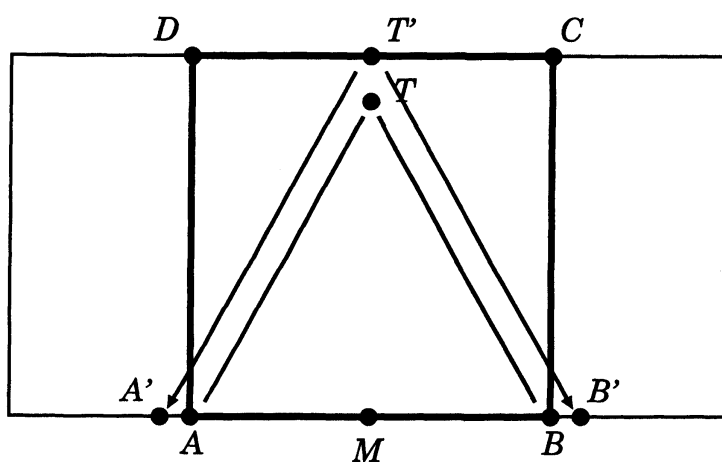
Figure 6: Sides of The Regular Triangle

Figure 7: Expansion of The Regular Triangle

To obtain the regular triangle of maximum size, we should expand the triangle ABT to a triangle with a top vertex T'. By the method explained in Sec.4.3, Parallel lines to TA and TB which start at T' can be obtained (Figure 7). And we obtain the regular triangle $A'B'T'$ at the center of the screen.

6 Conclusion

In this paper, we review the definition of a pattern and a pattern generation, and a pattern generation on the discretized screen. Next, we studied a correspondence between the discretized screen and cellular automata, and we studied the pattern generation on the cellular automata. Furthermore, we investigate a method to draw a line segment and a parallel line to a certain line as techniques for geometorical drawings. In the last part, we show an algorithm to draw a regular triangle of maximum size at the center of the screen.

Drawing a circle or an arc on two dimensional cellular automata is equal to drawing a circle or an arc on a paper by using a compass. Similarly, drawing a line segment on two dimensional cellular automata is equal to drawing a line segment on a paper by using a ruler. Therefore, we consider that a geometrical figure which can be drawn on a paper by using a

compass and a ruler can be drawn also on two dimensional cellualr automata.

Acknowledgements

reference

[1] Y.Mizuno, A New Solution of the Firing Squad Synchronization Problem for Two Dimensional Rectangle Arrays, *University of Aizu, Grad.Thesis.*, March, 2005.

[2] 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.

[3] A.Settle and J.Simon, Smaller solutions for the firing squad, *Theoretical Computer Science*, 276, pp.83-109, 2002.

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

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

[6] S.Watanabe and S.Okawa, Pattern Generation on Two Dimensional Cellular Automata, *Forum on Information Technology 2009*, A-023.

[7] J. Mazoyer, A six state minimal time solution to the firing squad synchronization problem, *Theoretical Computer Science*, vol.50, pp.183-238,1987.

[8] Y. Nishitani, Firing Synchronization Problems for Patterns specified by the Sub-General Position, *Cellular Automaton Seminar in University of Aizu* ,2010.

[9] S.Watanabe and S.Okawa, Circle Generation on Two-Dimensional Cellular Automata, *Algebraic Systems and Theoretical Computer Science, RIMS 1809*,pp161-168, 2012.

[10] S.Watanabe and S.Okawa, Rapid Circle Generation on Two-Dimensional Cellular Automata by $O(r^2)$ Time Algorithm, *Algebra and Computer Science, RIMS 1873*,pp163-169, 2014.

[11] A.Waksman, An optimum solution to the firing squad synchronization problem, *Information and Contorl*, pp66-78, 1966.