Formation of Vertical Dislocation Patterns in One-dimensional Computational Verb Cellular Networks

Xiaoli Lin and Tao Yang
Department of Electronic Engineering
Xiamen University
Xiamen, 361005, P.R. China
E-mails: linxiaoli616@gmail.com & taoyang.yangsky@gmail.com

***Abstract*—Computational verb cellular networks (CVCNs) are a new kind of cellular computational platform where the local rules are computational verb rules. In this paper, the mechanism of forming vertical dislocation patterns in one-dimensional CVCN is studied mathematically. The computational verb local rules (verb local rules, for short) of vertical dislocation pattern are constructed based on the decomposition of the global evolving pattern. Based on the verb local rules, mathematical equations are constructed to define the solutions of 1D CVCN. The solution is calculated numerically using Matlab to verify the usefulness of the theoretical predictions.**

Keywords-computational verb; cellular automata; 1D CVCN

I. Introduction

Cellular automata have been widely applied to many fields, such as sociology, biology, ecology, information science, mathematics, military science, and etc. It is usually recognized as a universal method of modeling. Computational verb cellular networks (CVCNs) generalize cellular automata by introducing dynamics into the construction of local rules. CVCN is a new kind of cellular computational platform where the local rules are computational verb rules.

The organization of this paper is as follows. In Section II, the architectures of 1D computational verb cellular networks will be presented. In Section III, some patterns found in 1D CVCN will be summarized. In Section IV, the vertical dislocation pattern solutions of 1D CVCN will be studied. In Section V, some concluding remarks will be included.

II. Architectures of 1D Computational Verb Cellular Networks

In a 1D CVCN [1], the cells are arranged along a line and each cell only has cells immediately to its left and right as its neighbors. Therefore, in a 1D CVCN the neighborhood of a center cell is arranged within a line segment. The verb local rules of 1D CVCN have the outputs of cells as antecedents. The reasoning results of verb local rules determine the state of a center cell.

In this paper, only 1D CVCNs with 1-neighborhood are studied. The verb local rules for 1-neighborhood is given by

$$\begin{aligned}&\text{IF } x_{i-1}(k)\ \mathsf{V}_{p,-1} \text{ AND } x_i(k)\ \mathsf{V}_{p,0} \text{ AND } x_{i+1}(k)\ \mathsf{V}_{p,1},\\ &\text{THEN } x_i(k+1)\ \widetilde{\mathsf{V}}_p;\ p=1,\ldots,m.\end{aligned} \tag{1}$$

Observe that the dynamics of the 1D CVCNs are determined by a set of m verb rules. If we cluster all dynamics of cells into a few computational verbs [2], which are called standard verbs henceforth, then we can build the local rules by exhausting all combinations of standard verbs. Let $S_{\mathsf{V}} = \{\mathsf{V}_1, \ldots, \mathsf{V}_n\}$ be the set of standard verbs. There are total n^3 possible combinations for a neighborhood and n^{n^3} possible 1D CVCNs constructed from S_{V}. When $n = 3$ the number is $n^{n^3} > 7.6256e{+}012$ and when $n = 4$ the number is $n^{n^3} > 3.4028e + 038$. Therefore, we only study the case when $n = 2$, which leads to $n^{n^3} = 256$ 1D CVCNs. When $n = 2$, we choose the standard verb set as $S_{\mathsf{V}} = \{\text{increase}, \text{decrease}\}$. When $n = 2$ the rule base in Eq. (1) is explicitly listed in Table I.

In this case, the verb local rules are listed in Table I. The verb reasoning of the eight verb rules in Table I results in

$$x_i(k+1) = \frac{\sum_{p=1}^{8} g_P f(x_{i+\alpha}(k)) \prod_{j=-1}^{1} S(\mathsf{V}_{p,j}, x_{i+j}(k))}{\sum_{p=1}^{8} \prod_{j=-1}^{1} S(\mathsf{V}_{p,j}, x_{i+j}(k))} \tag{2}$$

where $\mathsf{V}_{p,-1}$, $\mathsf{V}_{p,-1}$, $\mathsf{V}_{p,-1}$ are the computational verbs listed in the first, second and third column of the pth rule in Table I,

TABLE I. VERB LOCAL RULES OF THE 1D CVCN BASED ON A SET OF STANDARD VERBS ($S_{\mathsf{V}} = \{\text{increase}, \text{decrease}\}$).

$x_{i-1}(k)\ \mathsf{V}_{p,-1}$	$x_i(k)\ \mathsf{V}_{p,0}$	$x_{i+1}(k)\ \mathsf{V}_{p,1}$	$x_i(k+1)\ \widetilde{\mathsf{V}}_p$
decrease	decrease	decrease	$\widetilde{\mathsf{V}}_1$
increase	decrease	decrease	$\widetilde{\mathsf{V}}_2$
decrease	increase	decrease	$\widetilde{\mathsf{V}}_3$
increase	increase	decrease	$\widetilde{\mathsf{V}}_4$
decrease	decrease	increase	$\widetilde{\mathsf{V}}_5$
increase	decrease	increase	$\widetilde{\mathsf{V}}_6$
decrease	increase	increase	$\widetilde{\mathsf{V}}_7$
increase	increase	increase	$\widetilde{\mathsf{V}}_8$

respectively. g_P is a parameter to model the consequent verbs $\widetilde{\mathsf{V}}_p$. g_P is given by

$$g_P = \begin{cases} g_I, & \widetilde{\mathsf{V}}_p = \text{increase}; \\ g_D, & \widetilde{\mathsf{V}}_p = \text{decrease}. \end{cases} \tag{3}$$

The simplest verb similarities are given by

$$\begin{aligned} S(\text{increase}, x(k)) &= \frac{1}{1+e^{-\Delta_x/\Delta}} \\ S(\text{decrease}, x(k)) &= \frac{1}{1+e^{\Delta_x/\Delta}} \end{aligned} \tag{4}$$

where $\Delta > 0$, $k > 0$ and $\Delta_x = x(k) - x(k-1)$. $f(\cdot)$ is the output function. $\alpha = -1, 0$ and 1 is the directional influential index that defines the output of which neighbor influences $x_i(k+1)$. When $\alpha = -1, 0$ and 1, we call the 1D CVCN is left-influential, self-influential, and right-influential, respectively. Here we choose the nonlinear output function as

$$f(x) = \frac{1}{1+e^{-x}} \tag{5}$$

To avoid clutter and follow the convention of the established labeling method widely used in the study of cellular automata (CA), we use Wolfram notation to label 1D CVCNs when $S_{\mathsf{V}} = \{\text{increase}, \text{decrease}\}$. For example, corresponding to "rule 30 CA", the local rules of "rule 30 1D CVCN" is given by Table II. The last column in Table II lists a binary representation $\{b_p\}$ of elements in $S_{\mathsf{V}} = \{\text{increase}, \text{decrease}\}$. Here, when $\widetilde{\mathsf{V}}_p =$ decrease the corresponding $\{b_p\} = 0$ is denoted and when $\widetilde{\mathsf{V}}_p =$ increase the corresponding $\{b_p\} = 1$ is denoted. The binary number $b_8b_7b_6b_5b_4b_3b_2b_1$ is used as a unique identification number for a 1D CVCN. In the case shown in Table II this binary number is $b_8b_7b_6b_5b_4b_3b_2b_1 = 00011110 = 30$. Therefore, the 1D CVCN, of which the verb local rules are listed in Table II, is identified as "rule 30 1D CVCN".

III. Output Patterns of 1D CVCN

We found from many simulations that the output patterns of 1D CVCNs follow certain regularities with respect to their parameters listed as follows.

1. If $g_I = 1$ or -1 and $g_D = 1/g_I$, then $g_I = g_D$. So whatever rule is, the output patterns is the same as that shown in Fig. 1.

TABLE II. VERB LOCAL RULES OF "RULE 30 1D CVCN" BASED ON A SET OF STANDARD VERBS ($S_{\mathsf{V}} = \{\text{increase}, \text{decrease}\}$).

$x_{i-1}(k)\mathsf{V}_{p,-1}$	$x_i(k)\mathsf{V}_{p,0}$	$x_{i+1}(k)\mathsf{V}_{p,1}$	$x_i(k+1)\widetilde{\mathsf{V}}_p$	b_p
decrease	decrease	decrease	$\widetilde{\mathsf{V}}_1 =$ decrease	0
increase	decrease	decrease	$\widetilde{\mathsf{V}}_2 =$ increase	1
decrease	increase	decrease	$\widetilde{\mathsf{V}}_3 =$ increase	1
increase	increase	decrease	$\widetilde{\mathsf{V}}_4 =$ increase	1
decrease	decrease	increase	$\widetilde{\mathsf{V}}_5 =$ increase	1
increase	decrease	increase	$\widetilde{\mathsf{V}}_6 =$ decrease	0
decrease	increase	increase	$\widetilde{\mathsf{V}}_7 =$ decrease	0
increase	increase	increase	$\widetilde{\mathsf{V}}_8 =$ decrease	0

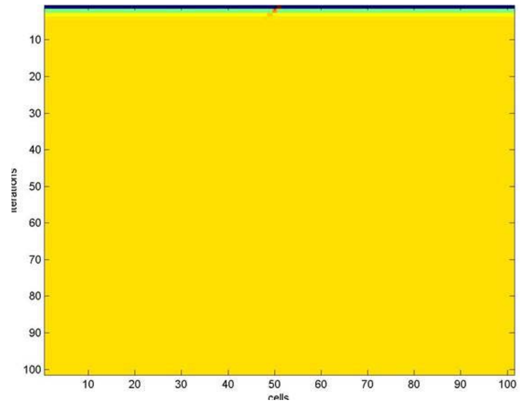


Figure 1. An output pattern when $g_I = 1$.

2. Stripe patterns and homogenous patterns have high probability in 1D CVCNs. When g_I approaches $-\infty$, 0 and $+\infty$, the patterns are most likely stripe. When g_I approaches 1 and -1, the patterns are most likely homogenous. If g_I is far from g_D, then big difference is between weights of "increase" and "decrease". This difference results in stripe patterns. An example of two-color stripe patterns is illustrated in Fig. 2.

3. If $\text{rule}_1 = \sim \text{rule}_2$ and $g_{I_1} = 1/g_{I_2}$, the two patterns must be the same as the rule pair shown in Fig. 3 of next page. The reason is as follows. If $\text{rule}_1 = \sim \text{rule}_2$, then $(b_8b_7b_6b_5b_4b_3b_2b_1)_{\text{rule}_1} = \sim (b_8b_7b_6b_5b_4b_3b_2b_1)_{\text{rule}_2}$. $g_8g_7g_6g_5g_4g_3g_2g_1$ is corresponding to $b_8b_7b_6b_5b_4b_3b_2b_1$. And the two sides of g_I and g_D are reverse to each other. But $g_{I_1} = 1/g_{I_2}$. Therefore, the patterns are the same.

IV. Solution of Vertical Dislocation Patterns

In [3], checkbox, flip-flop and homogenous patterns are discussed. In this paper, the solution of vertical dislocation pattern is studied mathematically.

Fig. 4(a) is the vertical dislocation patterns including two colors. It is found in self-influential 1D CVCN; namely, in the case of $\alpha = 0$ in Eq. (2). In this pattern, only the middle column is out of phase to the background, which is a two-color flip-flop pattern. We choose real numbers a, b stand for

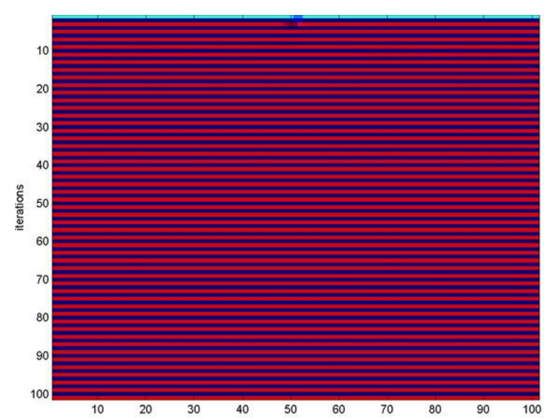


Figure 2. A two-color stripe pattern.

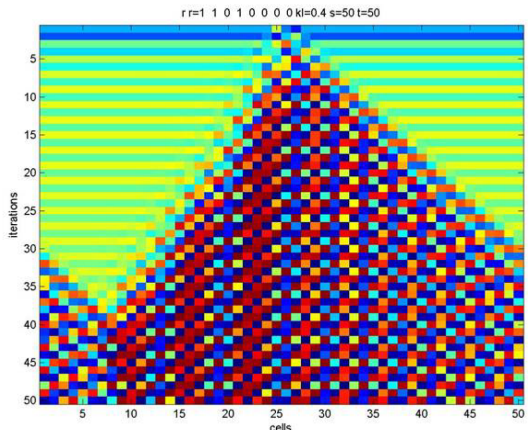
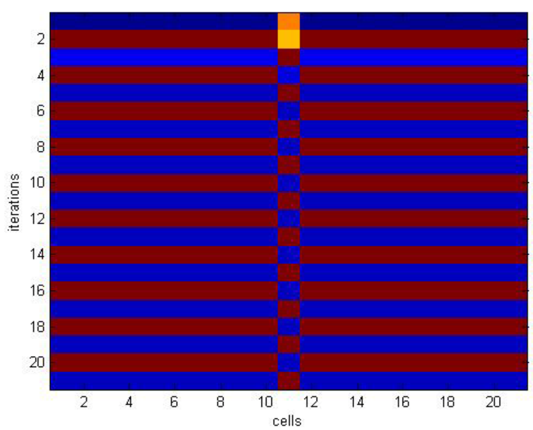
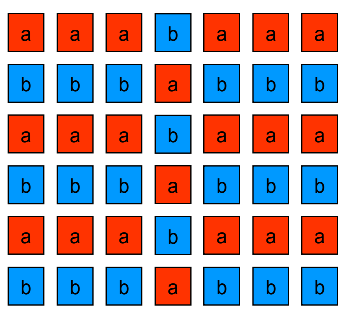


Figure 3. An output pattern when rule= 208, $g_I = 0.4$ or rule= 47, $g_I = 2.5$.



(a) The vertical dislocation pattern.



(b) Local patterns of six generation of cells to generated vertical dislocation pattern.

Figure 4. (a) The vertical dislocation pattern. (b) Local patterns of six generation of cells to generated vertical dislocation pattern.

two colors. Fig. 4(b) shows local patterns of six generation of cells to generated vertical dislocation pattern, of which the two colors are represented by a and b.

We analyze Fig. 4(b) and come out eight local evolving patterns, which are shown in Fig. 5 of next page, for the center cell to generate the vertical dislocation pattern. In each local pattern shown in Fig. 5, the first and the second row show the values of three cells at iterations $k-1$ and k, respectively. The third row shows the valued of the central cell at iteration $k+1$. To find the conditions of forming a vertical dislocation pattern in a 1D CVCN, we need to find the conditions under which all local patterns shown in Fig. 5 exist for a pair of constants a and b.

A. Local Evolving Pattern shown as Fig. 5(a)

For Fig. 5(a), the left cells, center cells and right cells have the following verb similarities.

$$S_{LI} \triangleq S(\text{increase}, x_{i-1}(k)) = \frac{1}{1+e^{-(b-a)}},$$
$$S_{LD} \triangleq S(\text{decrease}, x_{i-1}(k)) = \frac{1}{1+e^{(b-a)}}, \quad (6a)$$
$$S_{CI} \triangleq S(\text{increase}, x_{i}(k)) = \frac{1}{1+e^{-(b-a)}}.$$

$$S_{CD} \triangleq S(\text{decrease}, x_{i}(k)) = \frac{1}{1+e^{(b-a)}},$$
$$S_{RI} \triangleq S(\text{increase}, x_{i+1}(k)) = \frac{1}{1+e^{-(b-a)}}, \quad (6b)$$
$$S_{RD} \triangleq S(\text{decrease}, x_{i+1}(k)) = \frac{1}{1+e^{(b-a)}}.$$

It follows from Eq. (6) that

$$A \triangleq S_{LI} = S_{CI} = S_{RI}, B \triangleq S_{LD} = S_{CD} = S_{RD}. \quad (7)$$

Let y_o denote the output, then we have

$$y_o \triangleq f(x_{i+1}(k)) = \frac{1}{1+e^{-b}} \quad (8)$$

It follows from Eq. (2)$\sim$Eq. (5) that the state of the central cell in the next evolving step, $x_i(k+1) = a$, is calculated as follow.

$$\begin{aligned}
\sum_s &= S_{LD}S_{CD}S_{RD} + S_{LD}S_{CD}S_{RI} + S_{LD}S_{CI}S_{RD} \\
&+ S_{LD}S_{CI}S_{RI} + S_{LI}S_{CD}S_{RD} + S_{LI}S_{CD}S_{RI} \\
&+ S_{LI}S_{CI}S_{RD} + S_{LI}S_{CI}S_{RI} \\
&= (A+B)^3, \\
\frac{\sum_{sf}}{y_0} &= g_1 S_{LD}S_{CD}S_{RD} + g_2 S_{LD}S_{CD}S_{RI} \\
&+ g_3 S_{LD}S_{CI}S_{RD} + g_4 S_{LD}S_{CI}S_{RI} + g_5 S_{LI}S_{CD}S_{RD} \\
&+ g_6 S_{LI}S_{CD}S_{RI} + g_7 S_{LI}S_{CI}S_{RD} + g_8 S_{LI}S_{CI}S_{RI} \\
&= (g_2+g_3+g_5)AB^2 + (g_4+g_6+g_7)A^2B \\
&+ g_8 A^3 + g_1 B^3, \\
a &= \frac{\sum_{sf}}{\sum_s}.
\end{aligned} \quad (9)$$

B. Solution of Equation

We have designed MATLAB program to solve the equations in above symbolically. For each local evolving pattern, we can symbolically find the equations of output by using this program. For the example shown in Fig. 5(a), input is $[a\ a\ a\ b\ b\ b\ a]$, output is $f_1 = \sum_{sf} / \sum_s = a$ corresponding to Eq. (9).

Because vertical dislocation patterns have eight local evolving patterns, eight sets of equations will be established. The

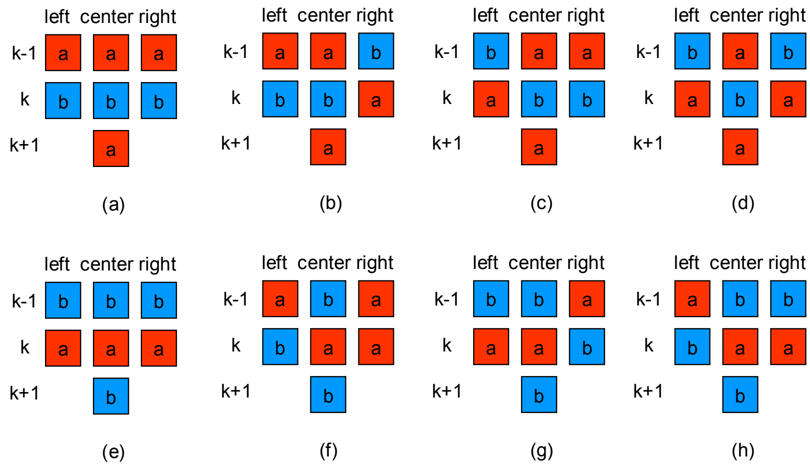
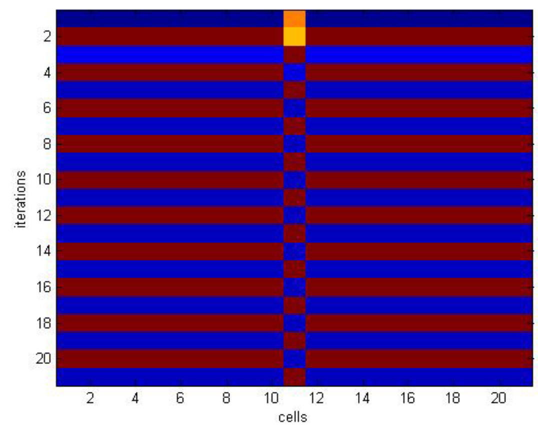


Figure 5. Eight local evolving patterns for Fig. 4(b).

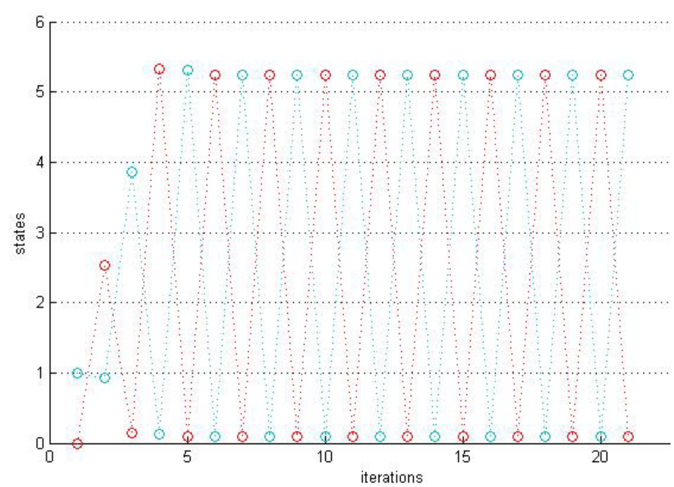
other seven sets of equations are found following the same procedure. In order to find the conditions under which vertical dislocation patterns form, we need to solve the eight equations by using Matlab.

Eight equations are all transcendental equation, and it is impossible to find analytical solutions. Instead, we can find some numerical solutions by Matlab. We get the following solution $a = 0.0998$, $b = 5.2492$, $\Delta = 0.5000$, $g_I = 10.0000$ and $g_D = 0.1000$. The simulation result of a 21-cell array is shown in Fig. 6. Toroidal boundary condition is used. The initial conditions are $x_i(-1) = 0$ and $x_i(0) = 0, i = 1, \ldots, 11$, except that the center cell has an initial state $x_{11}(0) = 1$.

Observe from Fig. 6(b) that a vertical dislocation pattern formed with the values in coordinate with our theoretical prediction. At the end of the evolution, all cells flip-flop between a and b.



(a) The vertical dislocation pattern.



(b) Local patterns of six generation of cells to generated vertical dislocation pattern.

Figure 6. (a) The vertical dislocation pattern. (b) Local patterns of six generation of cells to generated vertical dislocation pattern.

V. Conclusions

To analytically study the solutions of patterns generated by 1D CVCNs is extremely difficult. There are many complex patterns in 1D CVCNs, like chaos patterns, multiple patterns and ladder patterns. The more complex the pattern is, the more variables and equations will need. In order to solve these equations, we need to establish a more general solution and improve the computation speed.

References

[1] T. Yang, "Computational verb cellular networks: Part II-one-dimensional computational verb local rules," International Journanl of Computational Cognition, vol. 7, no. 1, pp. 36–38, March 2009.

[2] ——, "Using computational verbs to cluster trajectories and curves," International Journanl of Computational Cognition, vol. 4, no. 4, pp. 78–87, December 2006.

[3] ——, "Computational verb cellular networks: Part III-solutions of one-dimensional computational verb cellular networks," International Journanl of Computational Cognition, vol. 7, no. 2, pp. 1–9, June 2009.