Genaro

Collisions between particles induced by a regular language in complex elemental cellular automata

Genaro Juárez Martínez<sup>1</sup>

http://uncomp.uwe.ac.uk/genaro/

Unconventional Computing Group Faculty of Computing, Engineering and Mathematical Sciences University of the West of England Bristol, United Kingdom

1st International Workshop on Natural Computing - IWNC

December 14-15, 2006

<sup>&</sup>lt;sup>1</sup>genarojm@correo.unam.m×

#### Cellular Automata

Genaro

Preamble

Abstract Antecedent:

# Part I

### Preamble

### Abstract

#### Genaro

#### Preamble

Abstract Antecedents

We show the existence of a regular particle-based language in an elemental complex cellular automaton. We apply de Bruijn diagrams and tiles theory to determine the formal language and a full description of the evolution space characterized by triangles respectively. Particularity, we research Rule 110. Consequently, we propose a novel way to code initial conditions in Rule 110 and solving some interesting problems as: self-organization and the reproduction of a cyclic tag system. Finally, we discuss some others results and open problems in Rule 110.

#### ・ロト ・ 日本・ 小田・ ・ 田・ うらぐ

### Stages of most popular contributions

#### Genaro



#### 

#### Cellular Automata

Genaro

Background

Basic concept: Behavior

### Part II

### One-dimensional cellular automata

◆□▶ ◆□▶ ◆∃▶ ◆∃▶ → ヨ → のへぐ

### Cellular automata

Cellular automata are discrete dynamical systems evolving into an infinite regular lattice.

### Definition

A cellular automaton CA is a 4-tuple  $A = < \Sigma, u, \varphi, c_0 >$ evolving in *d*-dimension, where  $d \in \mathbb{Z}^+$ . Such that:

- $\Sigma$  represents the alphabet
- *u* the local connection, where,  $u = \{x_{0,1,\dots,n-1:d} | x \in \Sigma\}$ , therefore, *u* is a neigborhood
- $\varphi$  the local function, such that,  $\varphi: \Sigma^u \to \Sigma$
- $c_0$  the initial condition, such that,  $c_0 \in \Sigma^{\mathcal{Z}}$

Also, the local function induces a global transition between configurations:

$$\Phi_{\varphi}: \Sigma^{\mathcal{Z}} \to \Sigma^{\mathcal{Z}}.$$

#### Cellular Automata

#### Genaro

#### Background

Basic concepts Behavior

### Dynamics in one dimension



#### Cellular Automata

Genaro

#### Background

Basic concepts Behavior

### Wolfram's classes

Karel Culik II defines these classes in the next way:

Classes

- A CA is class I, if there is a stable state x<sub>i</sub> ∈ Σ, such that all finite configurations evolve to the homogeneous configuration.
- A CA is class II, if there is a stable state x<sub>i</sub> ∈ Σ, such that any finite configuration become periodic.
- ► A CA is class III, if there is a stable state, such that for some pair of finite configurations c<sub>i</sub> and c<sub>j</sub> with the stable state, is decidable if c<sub>i</sub> evolve to c<sub>j</sub>.
- Class IV includes all CA.

Thus, Class I  $\subset$  Class II  $\subset$  Class III  $\subset$  Class IV. Finally, Culik establishes that it is undecidable to determine the class of every CA. In particular, the *Elemental Cellular Automata* (ECA) named for Wolfram are all CA of order (k = 2, r = 1). Where  $|\Sigma| = k$  and u = 2r + 1.

### Karel Culik II and Sheng Yu

Undecidability of CA Classification Schemes

Complex Systems 2:177-190, 1988.

#### Cellular Automata

Genaro

#### Background

Basic concept Behavior

### Wolfram's classes

#### Cellular Automata



Figure: Wolfram's classes in elemental cellular automata. Initial random density of 0.5, to 414 cells in 238 times.

#### Cellular Automata

#### Genaro

#### Rule 110

Antecedents Dynamic in Rule 110 Regular language particles-based Applying the regular language More contributions in Rule 110

Discussion

Acknowledgements

The End

## Part III

### Rule 110

◆□▶ ◆□▶ ◆目▶ ◆目▶ 目 のへぐ

## History in Rule 110

The one-dimensional binary cellular automaton numbered Rule 110 in Stephen Wolfram's system of identification has been an object of special attention due to the structures or gliders which have been observed in evolution samples from random initial conditions. It has even been suggested that Rule 110 belongs to the exceptional class IV of automata whose chaotic aspects are mixed with regular behaviors; but in this case the background where the chaotic behavior occurs is textured rather than guiescent, a tacit assumption in the original classification. Whatever the merits of this classification. Rule 110 was awarded its own appendix (Table 15, see book below). It contains specimens of evolution including a list of thirteen gliders compiled by Doug Lind. Also, Wolfram presents the conjecture that the rule could be universal.



#### Stephen Wolfram

Theory and Aplications of Cellular Automata World Scientific Press, Singapore, 1986.



#### Wentian Li and Mats G. Nordahl

Transient behavior of cellular automaton rule 110 Physics Letters A 166:335-339, 1992.

#### Cellular Automata

#### Genaro

#### Antecedents

## History in Rule 110

In November 1998 at the Santa Fe Institute Matthew Cook demonstrates that Rule 110 is Universal! Simulating a novel cyclic tag system.



### Matthew Cook

Introduction to the activity of rule 110 (copyright 1994-1998 Matthew Cook)

http://w3.datanet.hu/~cook/Workshop/CellAut/Elementary/Rule110/ 110pics.html, January 1999.



Harold V. McIntosh

Rule 110 as it relates to the presence of gliders

http://delta.cs.cinvestav.mx/~mcintosh/oldweb/pautomata.html, January 1999.



#### Stephen Wolfram *A New Kind of Science* Wolfram Media, Inc., Champaign, Illinois, 2002.

#### Harold V. McIntosh Rule 110 Is Universal!

http://delta.cs.cinvestav.mx/~mcintosh/oldweb/pautomata.html, June 30, 2002.



### Matthew Cook

Universality in Elementary Cellular Automata *Complex Systems* 15/1:1-40, 2004.

#### Cellular Automata

#### Genaro

#### Rule 110

#### Antecedents

Dynamic in Rule 110 Regular language particles-based Applying the regular language More contributions in Rule 110

Discussion

Acknowledgements

The End

▲□▶ ▲□▶ ▲三▶ ▲三▶ 三三 のへぐ

### Rule 110

Rule 110 is an elemental cellular automaton. The local function determining the behavior is:

$$\begin{array}{ll} \varphi(0,0,0) \to 0 & \varphi(1,0,0) \to 0 \\ \varphi(0,0,1) \to 1 & \varphi(1,0,1) \to 1 \\ \varphi(0,1,0) \to 1 & \varphi(1,1,0) \to 1 \\ \varphi(0,1,1) \to 1 & \varphi(1,1,1) \to 0 \end{array}$$

Table: Evolution rule 110 – (01101110)<sub>2</sub>.



Figure: Random evolution in Rule 110. Initial density of 0.5, to 474 cells in 182 times.

#### Cellular Automata

#### Genaro

#### Rule 110

Antecedents Dynamic in Rule 110 Regular language particles-based Applying the regular language More contributions in

Discussion

Acknowledgements

### System of particles

Rule 110 have a complicated system of gliders. The figure show single, package and extensible gliders. Including the extensible glider gun!



#### Cellular Automata

#### Genaro

#### Dynamic in Rule 110

## System of particles

#### Genaro

structure	margins left - right				Vg	width	cap Rule 110 Antecedents	
	ems	oms	ems	oms				Regular language
er		1		1	2/3 pprox 0.666666	14	Т	particles-based Applying the regular
e <sub>l</sub>	1	•	1		-1/2 = -0.5	14	Т	language More contributions in
A	-	1	•	1	2/3 pprox 0.666666	6	Т	Rule 110
В	1		1	•	-2/4 = -0.5	8	Р	Discussion
<i>B</i> <sup>n</sup>	3	•	3		-6/12 = -0.5	22	Т	Acknowledgement
Â <sup>n</sup>	3		3		-6/12 = -0.5	39	Т	The Fact
<i>C</i> <sub>1</sub>	1	1	1	1	0/7 = 0	9-23	Р	The End
<i>C</i> <sub>2</sub>	1	1	1	1	0/7 = 0	17	Р	
<i>C</i> <sub>3</sub>	1	1	1	1	0/7 = 0	11	Р	]
<i>D</i> <sub>1</sub>	1	2	1	2	2/10 = 0.2	11-25	Р	
D <sub>2</sub>	1	2	1	2	2/10 = 0.2	19	Р	
E <sup>n</sup>	3	1	3	1	-4/15 $pprox$ -0.266666	19	Р	]
Ē	6	2	6	2	$-8/30 \approx -0.266666$	21	Р	
F	6	4	6	4	$-4/36 \approx -0.111111$	15-29	Р	]
G <sup>n</sup>	9	2	9	2	$-14/42 \approx -0.333333$	24-38	Р	
H	17	8	17	8	-18/92 pprox -0.195652	39-53	Р	
glider gun	15	5	15	5	-20/77 ≈ -0.259740	27-55	Р	]

Table: Properties of each glider in Rule 110.

◆□▶ ◆□▶ ◆目▶ ◆目▶ 目 のへぐ

### Formal languages

The formal languages theory provides a way to study sets of chains from a finite alphabet. The languages can be seen as inputs for some classes of machines or as the final result from a typesetter substitution system i.e., a generative grammar into the Chomsky's classification.

language	structure
recursively enumerated	Turing machine
context sensitive	linear bounded automata
context free	pushdown automata
regular	finite automata

#### Cellular Automata

#### Genaro

#### Rule 110

Antecedents Dynamic in Rule 110

#### Regular language particles-based

Applying the regular language More contributions in Rule 110

Discussion

Acknowledgements

The End

#### Lyman P. Hurd

Formal Language Characterizations of Cellular Automaton Limit Sets *Complex Systems* 1:69-80, 1987.

#### Stephen Wolfram

#### Computation Theory on Cellular Automata

Communication in Mathematical Physics 96:15-57, November 1984.



### 

### Formal languages

The basic model necessary for the languages of these machines (and for all computation), is the Turing machine. The machines recognizing each family of languages are described as a Turing machine with restrictions. In general, a *sequential machine* is a finite automata and in particular a regular language can be recognized by a finite automaton, a device with a finite number of internal states and with state transitions labeled by symbols from a finite alphabet. Thus, the sets of *regular expressions* on an alphabet are defined recursively as:

- 1.  $\phi$  is the regular expression representing the empty set.
- 2.  $\epsilon$  is the regular expression describing the set  $\{\epsilon\}$ .
- 3. For each symbol  $x \in \Sigma$ , x is a regular expression depicting the set  $\{x\}$ .
- If x and y are regular expressions representing languages Σ<sub>i</sub> and Σ<sub>j</sub> respectively, then (x + y), (xy), and (x\*) are regular expressions representing Σ<sub>i</sub> ∪ Σ<sub>j</sub>, Σ<sub>i</sub>Σ<sub>j</sub> and Σ<sup>\*</sup><sub>i</sub> respectively.

When it is necessary to distinguish between a regular expression x and the language determined by x, we shall use  $L_x$ .

#### Mats G. Nordahl

Formal languages and finite cellular automata

*Complex Systems* 3:63-78, 1989.

#### Cellular Automata

#### Genaro

#### Rule 110

Antecedents Dynamic in Rule 110

#### Regular language particles-based

Applying the regular language More contributions in Rule 110

Discussion

Acknowledgements

### Finite state machine



Figure: The first machine can read any word of language  $L_{R110}^+$  but only accepts ether configurations. The second machine only reads and accepts ether and A gliders. Nevertheless, the limitation is that both machines only can identify a phase for every particle.

#### Cellular Automata

#### Genaro

#### Rule 110

Antecedents Dynamic in Rule 110

#### Regular language particles-based

Applying the regular language More contributions in Rule 110

Discussion

Acknowledgements

For an one-dimensional cellular automaton of order (k, r), the de Bruijn diagram is defined as a directed graph with  $k^{2r}$  vertices and  $k^{2r+1}$ edges. The vertices are labeled with the elements of the alphabet of length 2r. An edge is directed from vertex i to vertex j, if and only if, the 2r - 1 final symbols of i are the same that the 2r - 1 initial ones in j forming a neighborhood of 2r + 1 states represented by  $i \diamond j$ . In this case, the edge connecting i to j is labeled with  $\varphi(i \diamond j)$ . The connection matrix M corresponding with the de Bruijn diagram is as follows:

$$M_{i,j} = \begin{cases} 1 & \text{if } j = ki, ki+1, \dots, ki+k-1 \pmod{k^{2r}} \\ 0 & \text{in other case} \end{cases}$$
(1)

#### Harold V. McIntosh

Linear cellular automata via de Bruijn diagrams

http://delta.cs.cinvestav.mx/~mcintosh/oldweb/pautomata.html, 1991.

#### Burton H. Voorhees

Computational analysis of one-dimensional cellular automata World Scientific Series on Nonlinear Science, Series A, Vol. 15, 1996.

#### Cellular Automata

#### Genaro

#### Rule 110

Antecedents Dynamic in Rule 110

#### Regular language particles-based

Applying the regular language More contributions in Rule 110

Discussion

**Acknowledgements** 

The End

・ロト ・ 日・ ・ 田・ ・ 日・ ・ 日・

Paths in the de Bruijn diagram may represent chains, configurations or classes of configurations in the evolution space.



Figure: de Bruijn diagram for Rule 110.

Now we must discuss another variant where the de Bruijn diagram can be extended to determine greater sequences by the period and the shift of their cells in the evolution space in Rule 110. A problem is that the calculation of extended de Bruijn diagrams grows exponentially with order  $k^{2r^n} \forall n \in \mathbb{Z}_+^+$ , the period and  $k^{2r^n} \forall n \in \mathbb{Z}_+^+$ .

#### Cellular Automata

#### Genaro

#### Rule 110

Antecedents Dynamic in Rule 110

Regular language particles-based

Applying the regular language More contributions in Rule 110

Discussion

Acknowledgements

The extended de Bruijn diagrams<sup>2</sup> calculate all the periodic sequences by the cycles defined in the diagram. These ones also calculate the shift of a periodic sequence for a certain number of steps; thus we can get de Bruijn diagrams describing all the periodic sequences characterizing a glider in Rule 110.



Figure: de Bruijn diagram calculating A glider and ether.

<sup>2</sup>The de Bruijn diagrams were calculated with the NXLCAU21 system developed by McIntosh for NextStep operating system (OpenStep and LCAU21 to MsDos). Application and code source are available from: http://delta.cs.cinvestav.mx/~mcintosh/oldweb/software.html 📃

#### Cellular Automata

#### Genaro

#### Regular language particles-based

In the figure we have two cycles: a cycle formed by vertex 0 and a large cycle of 26 vertices which is composed as well by 9 internal cycles. The evolution of the right illustrates the location of the different periodic sequences producing the A glider in distinct numbers. Following the paths through the edges we obtain the sequences or regular expressions determining the phases of the A glider. For example, we have cycles formed by:

- I. The expression  $(1110)^*$ , vertices 29, 59, 55, 46 determining  $A^n$  gliders.
- II. The expression (111110)\*, vertices 61, 59, 55, 47, 31, 62 defining nA gliders with a  $T_3$  tile between each glider.
- III. The expression  $(11111000100110)^*$ , vertices 13, 27, 55, 47, 31, 62, 60, 56, 49, 34, 4, 9, 19, 38 describing ether configurations in a phase (in the following subsection we will see that it corresponds to the phase  $e(f_{1}_{-}1)$ ).

The cycle with period 1 represented by vertex 0 produces a homogenous evolution with state 0. The evolution of the right shows different packages of A gliders, the initial condition is constructed following some of the seven possible cycles of the de Bruijn diagram or several of them. We can select the number of A gliders or the number of intermediate tiles  $T_3$  changing from one cycle to another  $r_3 \rightarrow r_2 \rightarrow r_3 \rightarrow r_2 \rightarrow r_2$ 

#### Cellular Automata

#### Genaro

#### Rule 110

Antecedents Dynamic in Rule 110

#### Regular language particles-based

Applying the regular language More contributions in Rule 110

Discussion

Acknowledgements



Cellular Automata

Genaro

#### Rule 110

Antecedents Dynamic in Rule 110

#### Regular language particles-based

Applying the regular language More contributions in Rule 110

Discussion

Acknowledgements

The End

Figure: Patterns calculated by de Bruijn diagrams up to 10 generations. Calculating *A*, *B*, *C* and *D* gliders, tiles and several periodic meshes.

Tiles

A plane of tiles T is a countable family of closed sets  $T = \{T_0, T_1, ...\}$  covering the plane without intervals or intersections. Defined as a join of sets (called a mosaic T):

$$\mathcal{T} = \bigcup_{i=0}^{n} T_i \,\forall \, n \in \mathbb{Z}_0^+$$
(2)

The "plane" is the Euclidian plane  $\mathbb{Z} \times \mathbb{Z}$  in elementary geometry. Rule 110 covers the evolution space through different sets of triangles  $T_n \forall n \in \mathbb{Z}_0^+$ , where *n* represent the size of the triangle counting the cells in some of its internal sides.



Branko Grünbaum and G. C. Shephard

Tilings and Patterns

W. H. Freeman and Company, New York, 1987. B . A

#### Cellular Automata

#### Genaro

#### Rule 110

Antecedents Dynamic in Rule 110

#### Regular language particles-based

Applying the regular language More contributions in Rule 110

Discussion

Acknowledgements

## Phases f<sub>i-i</sub>

#### Cellular Automata



#### Rule 110

Antecedents Dynamic in Rule 110

Regular language particles-based

Applying the regular language More contributions in Rule 110

Discussion

Acknowledgements

The End



where  $1 \le i \le 4$ 

Figure: Determining phases in Rule 110.

### Phases f<sub>i-</sub>i

The phases represent the periodic sequences (regular expressions of each glider) of finite length in the de Bruijn diagram. It is important to indicate that an alignment of a phase determines a set of regular expressions and another alignment defines another set of them. Cook determines two measures in the evolution space: horizontal  $\frown^i$  and vertical  $\nearrow^i$ . We only determine the horizontal case  $f_{i-1}$ . Phases  $f_{i-1}$  have four sub-levels consequence of the phases in  $T_3$  tile and each phase can be aligned *i* times generating all the possible phases (right part).

 $\begin{array}{rcl} \mbox{phases level one } (\mathcal{F}_1) & \to & \{f_{1}.1, \ f_{2}.1, \ f_{3}.1, \ f_{4}.1\} \\ \mbox{phases level two } (\mathcal{F}_2) & \to & \{f_{1}.2, \ f_{2}.2, \ f_{3}.2, \ f_{4}.2\} \\ \mbox{phases level three } (\mathcal{F}_3) & \to & \{f_{1}.3, \ f_{2}.3, \ f_{3}.3, \ f_{4}.3\} \\ \mbox{phases level four } (\mathcal{F}_4) & \to & \{f_{1}.4, \ f_{2}.4, \ f_{3}.4, \ f_{4}.4\} \end{array}$ 

Table: Four sets of phases  $\mathcal{F}_i$  in Rule 110.

Variable  $f_i$  indicates the phase currently used where the second subscript *i* (forming notation  $f_{i-i}$ ) indicates that selected set  $\mathcal{F}_i$  of regular expressions.

#### Cellular Automata

#### Genaro

#### Rule 110

Antecedents Dynamic in Rule 110

#### Regular language particles-based

Applying the regular language More contributions in Rule 110

Discussion

Acknowledgements

### Phases $f_{i-i}$

Finally, our notation codifies initial conditions by phases is in the following way:

$$\#_1(\#_2, f_{i-1})$$

where  $\#_1$  represents the glider according to Cook's classification and  $\#_2$  the phase of the glider if it has a period greater than four. We must indicate that the arrangement by capital letters for the  $\#_2$  parameter into the OSXLCAU21 system<sup>3</sup> does not have a particular meaning; it is only used to give a representation at the different levels for phases with gliders of periods module four.

<sup>3</sup>The OSXLCAU21 system have implemented a special panel for code initial conditions with all possible phases to each glider. The application and source are available from: http://uncomp.uwe.ac.uk/ genaro/OSXCASystems.html. Also you can see a practical introduction explaining our system in http://uncomp.uwe.ac.uk/genaro/papers/ OSXLCAU21/OSXLCAU21.html

#### Cellular Automata

#### Genaro

#### Rule 110

(3)

Antecedents Dynamic in Rule 110

Regular language particles-based

Applying the regular language More contributions in Rule 110

Discussion

Acknowledgements

### Phases $f_{i}_{i}$

Now we determine the phases  $f_{i-1}$  for A and B gliders as the figure illustrates.  $T_3$  tiles determine a phase  $\#_1$ ; in the case of A and B gliders only a  $T_3$  tile is necessary to describe their structure. In all the others cases, at least two  $T_3$  tiles are needed.



Figure: Phases  $f_{i-1}$  for A and B gliders respectively.

#### Cellular Automata

#### Genaro

#### Regular language particles-based

・ロト ・ 日 ・ ・ 日 ・ ・ 日 ・ ・ つ へ ()

### Phases $f_{i-i}$

Following each phase initiated by every  $T_3$  tile, the phases  $f_{i-1}$  for the A glider are as follows:

- $A(f_{1}_{1}) = 111110$
- $A(f_{2}) = 11111000111000100110$
- $A(f_{3}_{1}) = 11111000100110100110$

In general for every structure with negative speed, the phase  $f_{4-1} = f_{1-1}$ , for this reason the phase is not written. Each periodic sequences defined by  $T_3$  tiles conserves the regular expression property when basic rules are applied. Therefore,  $\epsilon$ ,  $A(f_{1-1})$ ,  $A(f_{1-1})+A(f_{1-1})$ ,  $A(f_{1-1})-A(f_{1-1})$ ,  $A(f_{1-1})+A(f_{1-1})-A(f_{2-1})-A(f_{2-1})$  are regular expressions (we use '-' to represent the concatenation operation in our constructions). Let us remember the codification in phases, A indicates the glider  $(\#_1)$  and  $f_{i-1}$  indicates the phase. Also, all phases  $f_{i-1}$  for the B glider are:

- $\blacktriangleright B(f_{1-1}) = 11111010$
- $\blacktriangleright B(f_{2}) = 11111000$
- $\blacktriangleright B(f_{3}_{-}1) = 111110001001100100110$
- ▶  $B(f_{4}_{-}1) = 11100110$

#### Cellular Automata

#### Genaro

#### Rule 110

Antecedents Dynamic in Rule 110

#### Regular language particles-based

Applying the regular language More contributions in Rule 110

Discussion

Acknowledgements

The End

▲□▶ ▲□▶ ▲ 三▶ ▲ 三▶ - 三 - のへで

### Phases $f_{i-i}$

The complete subset of regular expressions  $\Psi_{R110}$  for each glider in Rule 110 (see Appendix of our paper), serves as input data for the OSXLCAU21 system. So, is available from a digital file in:

### http://uncomp.uwe.ac.uk/genaro/rule110/listPhasesR110.txt

Genaro Juárez Martínez, Harold V. McIntosh, Juan C. Seck Tuoh Mora and Sergio V. Chapa Vergara Determining a regular language by glider-based structures called *phases*  $f_{i}$ .1 in Rule 110 by publish in *Journal of Cellular Automata*, special issue dedicated to Harold V. McIntosh, 2007.

#### Cellular Automata

#### Genaro

#### Rule 110

Antecedents Dynamic in Rule 110

#### Regular language particles-based

Applying the regular language More contributions in Rule 110

Discussion

Acknowledgements

The End

・ロト ・ 日・ ・ 田・ ・ 日・ ・ 日・

### Subset diagram

How to validate our regular language  $L_{R110}$  based in particles?



Figure: Subset diagram in Rule 110. Any sequence of  $\Psi_{R110}$  must follow a way into of the subset diagram begin from the biggest state. Other relevant properties is that besides the diagram determines *Garden of Eden* configurations, two minimal sequences are: (101010)\* and (01010)\*.

Harold V. McIntosh One Dimensional Cellular Automata by publish, 2007

#### Cellular Automata

#### Genaro

#### Rule 110

Antecedents Dynamic in Rule 110

Regular language particles-based

Applying the regular language More contributions in Rule 110

Discussion

Acknowledgements

The End

▲□▶ ▲□▶ ▲ 三▶ ▲ 三▶ - 三 - のへで

### Self-organization and synthesis of particles

We have demonstrated that all gliders can be produced by a collision from others gliders.

Our second paper, show a full description of several collisions to each glider *synthesis of gliders in Rule 110*, some of them with natural gliders. Too, some very complicated and rare extensible gliders.

### http://uncomp.uwe.ac.uk/genaro/rule110/glidersRule110.html



Genaro Juárez Martínez, Harold V. McIntosh and Juan C. Seck Tuoh Mora

Production of gliders by collisions in Rule 110

Lecture Notes in Computer Science 2801:175-182, 2003

Genaro Juárez Martínez, Harold V. McIntosh and Juan C. Seck Tuoh Mora

#### Gliders in Rule 110

International Journal of Unconventional Computing, 2/1:1-49, January 2006.

・ロト ・ 日 ・ ・ 日 ・ ・ 日 ・ ・ つ へ ()

#### Cellular Automata

#### Genaro

#### Rule 110

Antecedents Dynamic in Rule 110 Regular language particles-based

Applying the regular language

More contributions in Rule 110

Discussion

**Acknowledgements** 

### Reproducing the function of cyclic tag system

As an advance of our present work, we show the cyclic tag system inside the evolution space of Rule 110. This incredible result is reconstructed using our regular language for Rule 110. You can find some differences from *A New Kind of Science*, because it has mistakes that do not allow a good reconstruction. The mistakes were clarified by Cook in November 2002 (personal communication).

### http://uncomp.uwe.ac.uk/genaro/rule110/ctsRule110.html

Writing the sequence 1110111 on the tape of the cyclic tag system and a leader component at the end with two solitons. Our reconstruction is developed over an evolution space of 56,240 cells in 57,400 generations, i.e., a space of 3,228,176,000 cells with a computer Pentium II to 233 mhz, operating system OpenStep and 256MB of RAM, February 2003. Collaborations of Harold V. McIntosh and Juan C. Seck Tuoh Mora.

Genaro Juárez Martínez, Harold V. McIntosh, Juan C. Seck Tuoh Mora and Sergio V. Chapa Vergara

Reproducing the cyclic tag systems developed by Matthew Cook with Rule 110 using the phases  $f_{i}.1$ 

preprint available from http://uncomp.uwe.ac.uk/genaro/papers.html

#### Cellular Automata

#### Genaro

#### Rule 110

Antecedents Dynamic in Rule 110 Regular language particles-based

Applying the regular language

More contributions in Rule 110

Discussion

Acknowledgements

### More contributions in Rule 110

### Universality



Turlough Neary and Damien Woods P-completeness of cellular automaton Rule 110 Lecture Notes in Computer Science 4051:132-143, July 2006.



### Mirko Rahn

#### Universalität in Regel 110

http://www.stud.uni-karlsruhe.de/~uyp0/uni110.foil.ps.gz, 18 März, 2003.

### Universality and cyclic tag systems



#### Kenichi Morita

Simplifying Universal One-Dimensional Reversible Cellular Automaton Infinite Configurations

Conference at the Automata 2006, Hiroshima University, 2006.



#### Kenichi Morita

Simple Universal One-Dimensional Reversible Cellular Automata Journal of Cellular Automata, by publish, 2007.

#### Cellular Automata

#### Genaro

#### Rule 110

Antecedents Dynamic in Rule 110 Regular language particles-based Applying the regular

More contributions in Rule 110

Discussion

Acknowledgements

The End

・ロト・西ト・ヨー うへぐ

### More contributions in Rule 110

### Gliders and tiles

Harold V. McIntosh A Concordance for Rule 110 http://delta.cs.cinvestav.mx/~mcintosh/oldweb/pautomata.html, 2000.



Genaro Juárez Martínez and Harold V. McIntosh ATLAS: Collisions of gliders like phases of ether in Rule 110 http://uncomp.uwe.ac.uk/genaro/papers.html, August 2001.



Genaro Juárez Martínez Introduction to Rule 110

Conference at the *Rule 110 Winter Workshop*, Bielefeld, Germany, March 2004. (available from http://uncomp.uwe.ac.uk/genaro/papers.html)

### Rule 110 objects

Genaro Juárez Martínez, Harold V. McIntosh, Juan C. Seck Tuoh Mora and Sergio V. Chapa Vergara Rule 110 objects and other construccions based-collisions *Journal of Cellular Automata*, in press, 2007.

4

<sup>4</sup>A full list of references, advances and related topics in Rule 110 is available from http://uncomp.uwe.ac.uk/genaro/rule110\_html  $\sim$ 

#### Cellular Automata

#### Genaro

#### Rule 110

Antecedents Dynamic in Rule 110 Regular language particles-based Applying the regular language

More contributions in Rule 110

Discussion

Acknowledgements

### Discussion

### Final commentaries

Several questions arise because it seems that the evolution of Rule 110 language should always be regular. For instance:

- How a regular language can be able of constructing a universal machine?
- Could Rule 110 determine new grammars?
- Could we project this language to two-dimensional finite-state automata?
- Could Rule 110 be able of implementing unconventional logic operations by glider-based reactions?
- Could Rule 110 support a Turing machine, universal constructor and self-reproduction? Also, with a system of gliders.

Well, it is only the beginning!

#### Cellular Automata

#### Genaro

#### Rule 110

Antecedents Dynamic in Rule 110 Regular language particles-based Applying the regular language More contributions in Rule 110

Discussion

Acknowledgements

### Acknowledgements

- ► EPSRC grant reference: EP/D066174/1
- Faculty of Computing, Engineering and Mathematical Sciences
- University of the West of England
- Important collaborations in Rule 110 research by Profs.
   Harold V. McIntosh and Juan Carlos Seck Tuoh Mora
- Personally to Profs. Katsunobu Imai and Andrew Adamatzky by invited me to present this talk in IWNC 2006

#### Cellular Automata

#### Genaro

#### Rule 110

Antecedents Dynamic in Rule 110 Regular language particles-based Applying the regular language More contributions in Rule 110

Discussion

Acknowledgements

### The End



#### Cellular Automata

#### Genaro

#### Rule 110

Antecedents Dynamic in Rule 110 Regular language particles-based Applying the regular language More contributions in Rule 110

Discussion

Acknowledgements

The End

Figure: Glider gun by collision among particles with OSXLCAU21 system. The regular expression is:  $e^+-D_1(C,f_3-1)-e^-C_1(A,f_1-1)-e^-\overline{E}(B,f_1-1)-e^+$ .