

APPLICATION OF SEMITOTALISTIC 2D CELLULAR AUTOMATA ON A TRIANGULATED 3D SURFACE

M. ZAWIDZKI
Ritsumeikan University, Japan

ABSTRACT

This paper presents the application of semi-totalistic, also called outer totalistic cellular automata on any three-dimensional (3D) surface. Cellular automata (CA) can be applied in controlling the state of a free-form building envelope. Such an intelligent ‘skin of a building’ can have any shape and a certain ‘organic’ appearance which can be dynamically controlled in response to the changes of the external conditions or users’ requirements. Any 3D surface can be triangulated. This means that it can become a grid of topologically identical elements. With the exception of boundary conditions, applicable to the elements positioned at the edge of the surface or around holes, every triangle in the grid has exactly three neighboring triangles. As with a CA, every element of a triangulated surface can be individually assigned with characteristics such as color or transparency level, which is analogous to a CA ‘state’. Therefore, it is possible to control to some degree the state of the whole surface taking advantage of the emergent behavior of the CA. The concept of CA on a triangular tessellation is discussed followed by discussion about the entities of irregularity of a grid, called ‘holes’ and ‘edges’. The concept of an ‘organic’ pattern in the context of CA is also briefly discussed. Two-dimensional (2D) CA in the triangular tessellation is discussed and implemented. A brief study covers the entities of neighborhood, type of rules (semi-totalistic and general), rule encoding and search for rules that meet given criteria. An implementation is performed on a regular triangular grid, irregular triangular grid and an imported triangulated 3D mesh which is irregular and has holes. A selected 2D triangular CA is applied on an imported triangulated 3D model.

Keywords: cellular automaton, triangulated surface, triangular CA, skin of a building.

1 INTRODUCTION

A ‘building envelope’ or a ‘skin of a building’ is a major engineering and aesthetic development in architecture. This interface between the exterior and interior of a building serves several functions, such as protection from external factors, as it improves security and reduces the levels of noise and pollution, protection from climate changes (temperature, humidity, glare), provision of natural light and visual contact with the environment, as well as isolation from it. A building envelope can also help to save energy (insulation, ventilation, thermal mass: heat/cool storage) or even actively collect energy (e.g. solar energy – by photovoltaics or solar–thermal collectors). It also allows adjusting the appearance to the environment or the customer’s preference. The variability of the external/environmental conditions and the need to create certain desired conditions inside buildings demand controlling the physical properties of this interface. Such a requirement is rational and desirable, although in reality very difficult. This paper demonstrates a new way of statistical control of the state (opacity or color) of elements of a building envelope modeled by a triangulated surface. A potential application of this method is shading of buildings.

The state of the surface is controlled by the application of the nature-inspired cellular automata (CA). The concept of CA replaces the centralized and direct control over the state of the elements of the surface, by emergence resulting from local interactions among these elements. The patterns generated in this way often demonstrate fascinating ‘freedom’ and ‘reason’ [1] and can be pleasing to the human eye.

Any three-dimensional (3D) surface can be triangulated (Fig. 1) that is approximated by elements of the same topology (triangular faces). As each element has the same number of neighboring elements



Figure 1: Geometrically complex but topologically uniform cellular surface of a roof. Every individual triangular facet of the surface of the demonstrated building envelope has three immediate neighbors.

(with the exception of the elements at boundaries) – the use of CA seems intuitive and due to modularization (every cell is the same), can be fabricated inexpensively [2].

2 SKIN

‘Skin of a building’ is a concept in engineering and architecture which reflects the complexity of functions performed by this type of spatial barrier. A comparison between the skin of the mammals or other animals and the building envelope is shown in Table 1 [3–6].

The table shows that the physical properties and functions of a building envelope are indeed analogous to those of the skin. Both skin and a building envelope are multi-layered complex composites made to face the constantly changing environmental conditions. Until today it seems that the organic skin outperforms the man-made building facades. The use of a biomimetic – cellular composition on the surface of the building envelope allows the implementation of dynamic changes of its state in a new way – based on a CA. The ability of a building facade to dynamically respond to the changing conditions is assumed to be an ‘organic’ quality. Moreover, as all cells composing such a surface are topologically identical, they can be mass produced inexpensively [2].

3 ARE CELLULAR AUTOMATA ORGANIC?

CA are dynamical systems which are discrete in space and time, operate on a topologically regular lattice and are characterized by local interactions. CA models are used in computability theory, mathematics, physics, complexity science, theoretical biology and microstructure modeling. The most important quality of the CA is the emergence: a collection of very simple cells may produce complex function (Fig. 2).

Table 1: A comparison between the functions of skin and building envelope.

Skin	Building envelope
<i>Protection:</i> the human skin is an anatomical barrier protecting the human organism from pathogens of the external environment. The human skin acts as a large organ of body defense. The Langerhans cells in the skin are part of the adaptive immune system [3, 4].	Outer shell to protect the indoor environment.
<i>Sensation:</i> the human skin contains a variety of nerve endings that react to heat and cold, touch, pressure, vibration and tissue injury.	Control of air pressure boundaries of sorts. Sensory contact with the outside (mostly visual, also audial).
<i>Heat regulation:</i> The rich blood supply of the human skin (far greater than its requirements) has the capacity of precisely controlling the energy loss by radiation, convection and conduction. Dilated blood vessels can increase perfusion and heat loss, while constricted vessels greatly reduce cutaneous blood flow and conserve heat.	Temperature control (insulation, heat transfer, thermal mass).
<i>Control of evaporation:</i> A relatively dry and semi-impermeable barrier to fluid loss [4].	Moisture control.
<i>Storage and synthesis:</i> storage center for lipids and water.	Heat storage (Thermal mass, Trombe wall [5]).
<i>Absorption:</i> Oxygen, nitrogen and carbon dioxide can diffuse into the epidermis in small amounts, and some animals use their skin as their sole respiration organ [6].	Ventilation (indoor air quality, hygiene and public health).
<i>Water resistance:</i> The human skin acts as a water resistant barrier so that essential nutrients are not washed out of the body.	Water barrier (water condensation).
<i>Pigmentation</i> (camouflage, mimicry, UV protection, communication, sexual reproduction, warning etc). For example, erector pili muscles in the animals have the purpose of changing the appearance, making animals look larger. This is an involuntary muscle action that happens when mammals are in danger or about to fight.	Appearance (aesthetics, communication).
Other animal coverings such the arthropod exoskeleton or the seashell have different developmental origin, structure and chemical composition.	Structural integrity (shell).

In the case of the CA, the rules determining the behavior of individual cells are usually the same and often very simple. The interactions between these cells, however, can produce patterns of unexpected complexity. In nature, the mechanisms are more subtle, but the general concept seems to be similar. In the design practice, the term ‘organic’ has a broader meaning than ‘of, relating to, or derived from living organisms’ [8]. It is often used to describe any naturally occurring phenomenon, for example, a snowflake, which in a scientific sense is inorganic (Table 2).

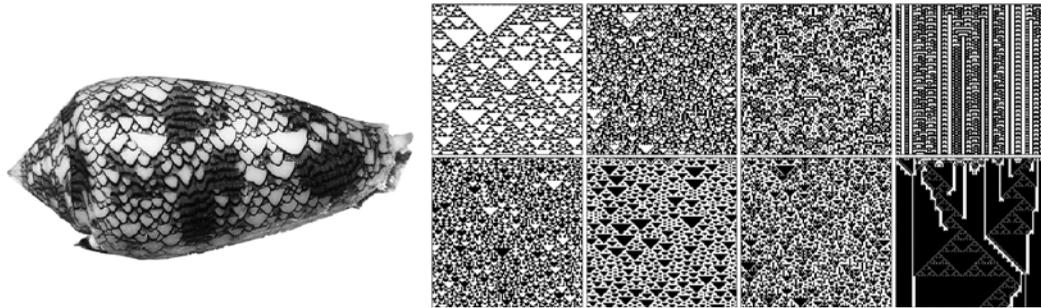
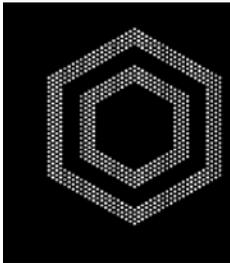
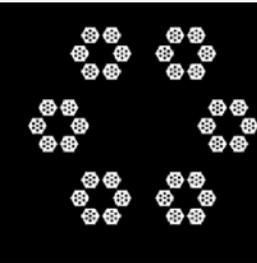
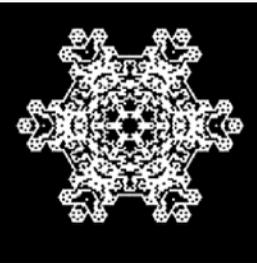


Figure 2: On the left, a mollusk shell (*Conus textile*, photo © 2005 Richard Ling), on the right seven examples of patterns generated by elementary cellular automata (EA) and one by a 3-color range-2 1D CA. A mollusk shell, like a 1D CA grows one line at a time, with new shell material being produced by a lip of soft tissue at the edge of the animal inside the shell [7].

Table 2: Man-made versus organic appearance.

Man-made (non-organic)		Natural (organic)	
			
2-color hexagonal CA #3629 at 43 step	2-color hexagonal CA #1126 at 43 step	2-color hexagonal CA #12837 at 44 step	A real snowflake (Photo © 2010 Northwoods Press)
Simple pattern	Nested pattern	Organic pattern	Natural object
Geometry: Simple	Nested	Complex	Complex + stochastic
Perception: Trivial	Complicated, but easily understandable	Complicated, not easily understandable but intuitively rational	Complicated, rational and imperfect
Structure: Global shape has no interactions among the cells on the lower level	Nested shape, the same structure is repeated at every level, no interactions among cells	The shape emerges from multiple simple interactions at the cellular level	The shape emerges from not-so-simple multiple interactions at the cellular level

Since CA is a biomimetic concept, its emergent behavior represented graphically by patterns has also some organic quality. There are four main classes of CA behavior [7], as distinguished and shown in Fig. 3. For each one of these classes there are cases which can be considered organic. Class 4 can be clearly identified as the one producing organic patterns.

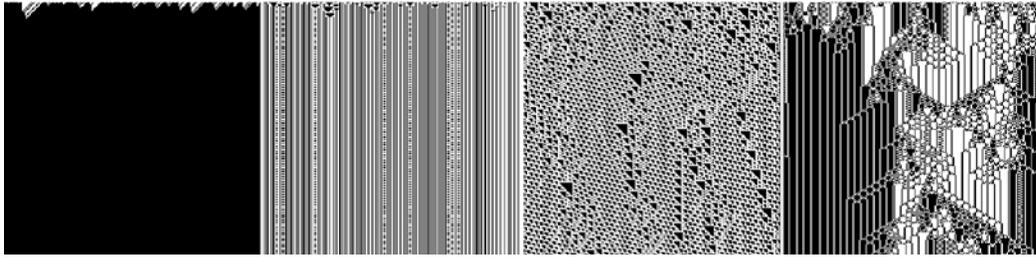


Figure 3: Four classes of CA behavior: 1) Constant, 2) Periodic, 3) (Pseudo) Random, 4) Complex. Although some are more visually interesting than others, all have certain organic qualities. While it is questionable in class 1, 2, and 3, it is clear that class 4 is organic.

CA is a decentralized system, where all the cells respond only to their neighbors and their state cannot be altered ‘manually’ (besides their initial conditions which are set directly). Usually the state of the cells can be predicted only to a certain degree, for example, the average density. Although stochasticity can also be implemented, regular CA are deterministic, but in most cases (the 3rd and 4th classes of behavior) due to their computational irreducibility, the state of a certain cell can only be known by running an experiment. For applications such as building shading this level of control is sufficient [1]. The results of the application of a CA over a given (in this case triangulated, but it can be any topologically regular) mesh are presented in Table 3.

Table 3: Application of a CA on a triangulated surface.

The CA characteristics	The characteristics of the surface with a CA
For a given CA, the state controlling unit at every cell is identical.	The control units can be mass-produced for the whole surface.
Every cell is interconnected, but can be programmed independently.	It is possible to apply different CA in various areas of the same surface.
The system is decentralized.	The system is robust. Failure of units stays localized, does not influence the rest of the cells.
Computational irreducibility	The state of cells of class 3 and 4 CA can be predicted only statistically. A certain degree of uncertainty may be desirable for aesthetic reasons.
Determinism	It is possible to run a simulation for a given initial conditions in order to determine the final state of cells.

4 FROM A TRIANGULATED SURFACE TO A TRIANGULAR GRID

The computer model of a triangulated surface uses a common convention to operate on a large number of graphic primitives: a list of vertices’ coordinates and a list with sub-lists of the vertices of the polygons (in this case triangles). Triangles are oriented – some of them are pointing ‘up’ and some ‘down’, as shown in Fig. 4. The colors and other characteristics of the polygons can be stored as an additional list.

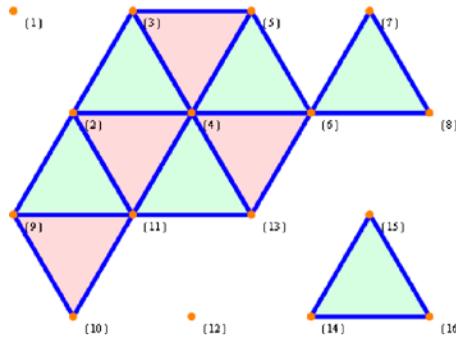


Figure 4: An example of a triangular grid with oriented triangles.

5 A CELLULAR AUTOMATON ON A TRIANGULAR LATTICE: THE TRIANGULAR CA
 Although the most commonly used lattice for CA is an orthogonal grid (1D, 2D or 3D), several studies have been carried out to test the properties of other tessellations such as triangular, hexagonal [9–11]. The operation on a triangular grid works on both regular and irregular grids, as shown in Fig. 5. The same procedure can be applied on regular and deformed triangles of a surface of any shape and size. The only difference is the coordinates of the vertices, while the topological information about the graphic primitives remains the same.

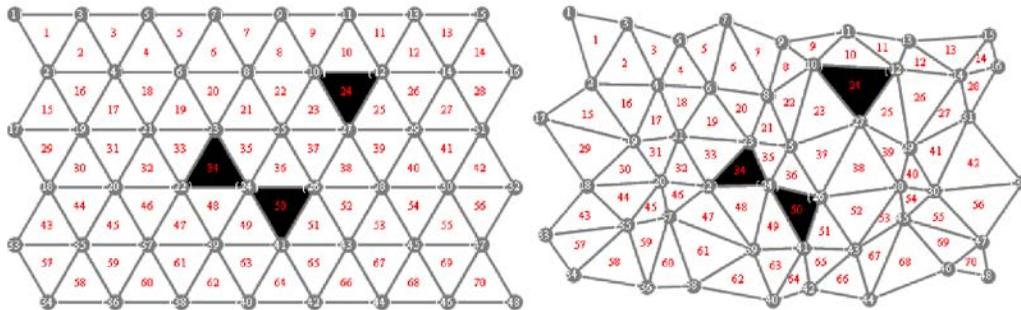


Figure 5: The same topology for regular and irregular triangular meshes.

6 SEMITOTALISTIC 2-COLOR 2-DIMENSION RANGE-1 TRIANGULAR CELLULAR AUTOMATA (ST2C2DR1TCA)

For a triangular lattice the smallest neighborhood is 1, which means that every cell has a neighborhood composed of cells adjacent to the three sides of the cell (von Neumann neighborhood = 3 neighbor cells) or adjacent to the sides and nodes (Moore neighborhood = 12 neighbor cells), as shown in Fig. 6.

In the case of a general CA, the information from every cell is taken, for a totalistic rule – the sum of all the values (the main + neighboring cells), for semi-totalistic rules – the value in the main cell and the sum of the values in the neighboring cells. The size and type of the neighborhood and the type of the CA determine the number of all possible rules. Some examples are shown in Table 4.

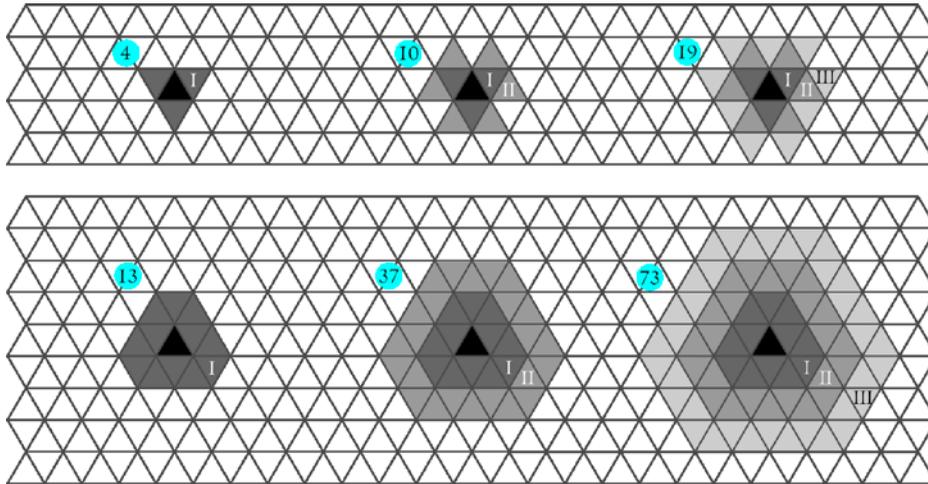


Figure 6: On the top: von Neumann's neighborhood, on the bottom: Moore's neighborhood for a triangular lattice. From left to right: ranges 1, 2 and 3. Values in the circles indicate the total number of cells for a given neighborhood.

Table 4: Numbers of possible rules for various types of 2-color triangular cellular automata (TCA).

	von Neumann	Moore
Range-1		
General	65536*	$\sim 10^{2466}$
Semi-totalistic	256 [†]	$\sim 10^7$
Totalistic	32 [‡]	16384*
Range-2		
General	$\sim 10^{308}$	$\sim 10^{413732481637}$
Semi-totalistic	$\sim 10^6$	$\sim 10^{22}$
Totalistic	2048*	$\sim 10^{11}$
Range-3		
General	$\sim 10^{157826}$	$\sim 10^{28431479646764985973578}$
Semi-totalistic	$\sim 10^{11}$	$\sim 10^{44}$
Totalistic	$\sim 10^6$	$\sim 10^{22}$

In this project, [†] indicates range-1 semi-totalistic TCA that were analyzed, [‡] indicates range-1 totalistic TCA with von Neumann neighborhood that were omitted and * indicates other types of TCA worth considering due to the manageable number of possible rules.

The totalistic triangular cellular automata (totalistic TCA) with von Neumann neighborhood are too simple to be used for controlling the state of a triangulated surface, hence the analysis starts from the semi-totalistic TCA. In this case, the initial state of every cell is given as a two-element list containing information about the state of the main cell (0 or 1) and the sum of the values of the neighboring cells (possible values: 0, 1, 2, 3). There are $2^{(2^4)} = 256$ semi-totalistic 2-color 2-dimension range-1 triangular

cellular automata (ST2C2DR1TCA). Table 5 shows rule #107 ($01101011_2=107_{10}$). The color of all three neighbor cells is uniform and represents the mean value of the neighbors. The values of 1, 2/3, 1/3 and 0 are represented by different shades of gray: 100%, 66%, 33% and 0% of black, respectively.

On the triangular lattice there are two distinctive configurations of the neighborhood depending on the orientation of the main cell as shown in Fig. 7.

To avoid (visual) confusion while analyzing the emerging patterns, in all the CA computations non-periodic boundary conditions were applied.

Table 5: An example of a semi-totalistic triangular cellular automaton, rule #107.

	{1,3} → 0	
	{1,2} → 1	
	{1,1} → 1	
	{1,0} → 0	
	{0,3} → 1	
	{0,2} → 0	
	{0,1} → 1	
	{0,0} → 1	

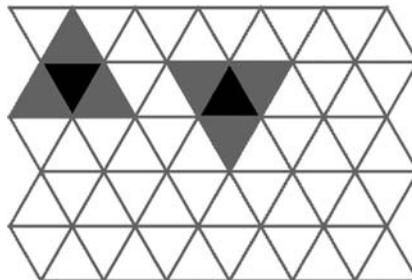


Figure 7: An example of two orientations of a cell with a range-1 neighborhood. The main cells are indicated in black, neighbor cells in gray.

7 THE SEARCH FOR APPROPRIATE RULES

The criteria for an ‘appropriate’ rule which can be applied for shading of a building envelope:

- non-constant: the pattern on the array differs at any two consecutive steps (until the evolution stops), which means the evolution does not soon ‘die out’, nor does it reach a ‘fixed’ point soon after the first step (as in the 1st class CA).

- non-blinking: cells which are not directly affected by the evolution of a CA (lie within the neighborhood of an active cell) do not change their state.
- stable: the cells do not change their state after all the cells within the range of the CA have been subject to the CA evolution.

CA are in principle deterministic and a number of ways to program them to perform preconceived actions were studied [12]. Nevertheless, often due to their computational irreducibility [7] the procedure turns to a series of trial-and-error experiments. Some of the criteria mentioned above can be formalized as shown in Table 6 (see Appendix).

Figure 8 shows the 5th step of evolution of all the CA generated according to Table 6 starting from the same initial conditions (a single black cell in the 3rd row and 5th column).

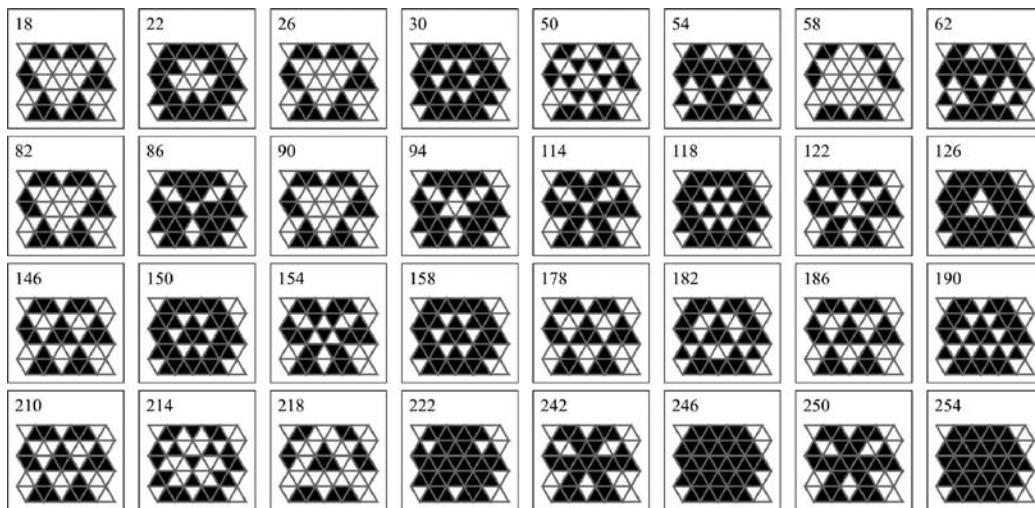


Figure 8: All 32 ST2DR1TCA meeting the preliminary criteria after five steps of evolution, starting from the same initial conditions.

The search for the most appropriate TCA was done by analyzing the generated patterns. The rules which patterns differ excessively at one step were filtered out. The selection mechanism was based on subtracting the values at the 2nd step from the values in the 3rd step of evolution and adding up the absolute values. This gave the total number of all the cells that differ between these two steps. Figure 9 shows the 2nd and 3rd step of evolution of all preselected TCA and the number of different cells Δ .

The values of Δ for every even rule out of 256 are visualized below in Fig. 10. The odd rules (with 1 at the first binary position) are excluded, because $\{0,0\} \rightarrow 1$ means that every white cell surrounded by white cells will turn black, which results in a very high value of Δ .

The patterns of six selected rules (from the green area of the previous graphics) were further analyzed. The next comparison of patterns is shown in Fig. 11.

Due to blinking (never stabilizes), rule 122 is not appropriate for shading purposes. The remaining five TCA (114, 242, 250, 246, 254) can be applied for shading the triangulated surface. The stabilization time is comparable. The final selection of the TCA can be based on the desired approximate density at the stabilized state (58, 77, 84, 99 or 100% black). For the final presentation in this paper,

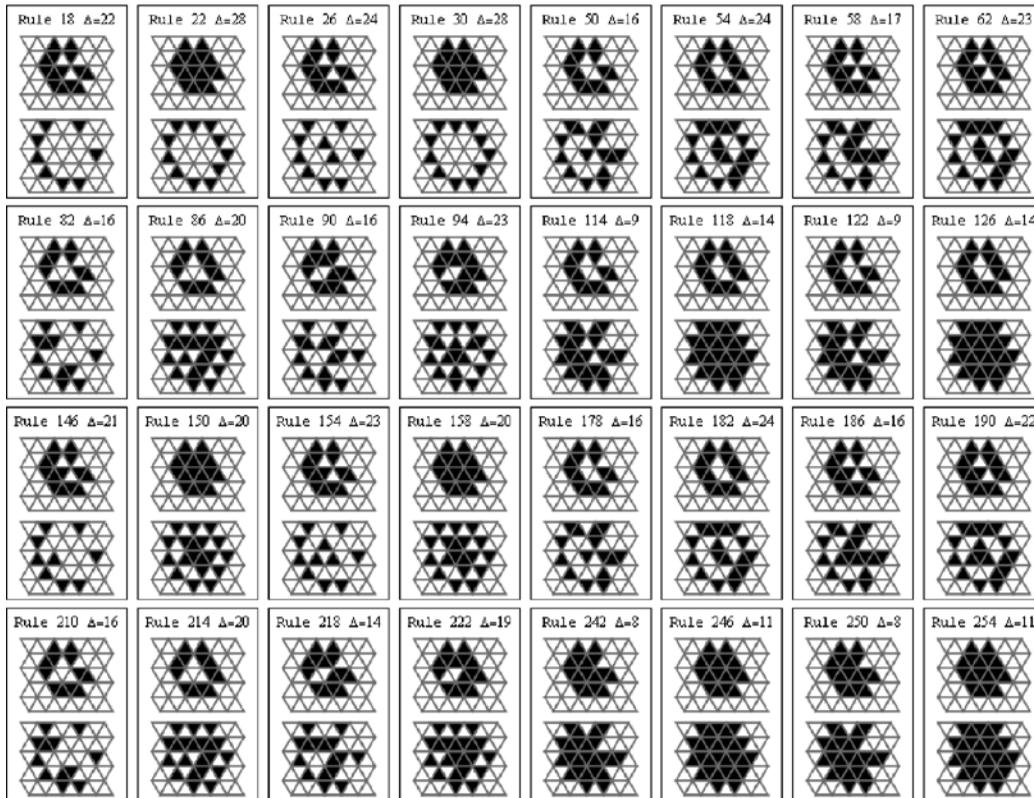


Figure 9: Some rules produce very different patterns at each step, not a desirable quality in this project; the initial conditions as before, a single black cell in the middle of the array.

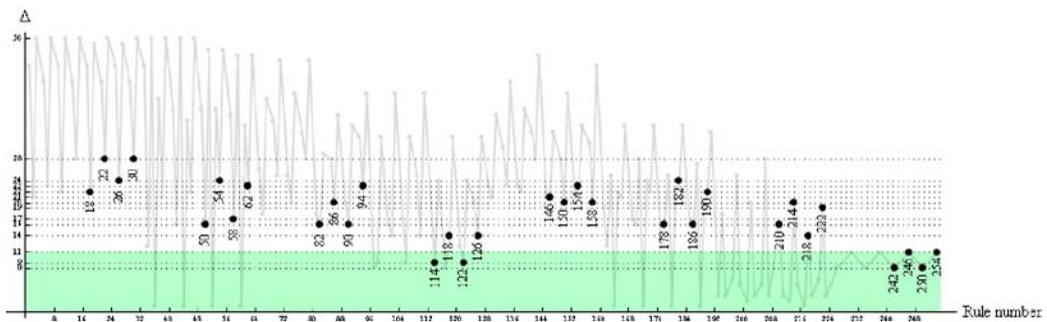


Figure 10: The TCA rules under consideration are indicated with black dots. The initial conditions are: a cluster of three black cells in the middle of an array. In the background all the non-constant rules are indicated. The green area indicates an arbitrary threshold for a rule 'stability'. There are some rules which seem more 'stable' but it is irrelevant, since they do not meet other criteria mentioned above.

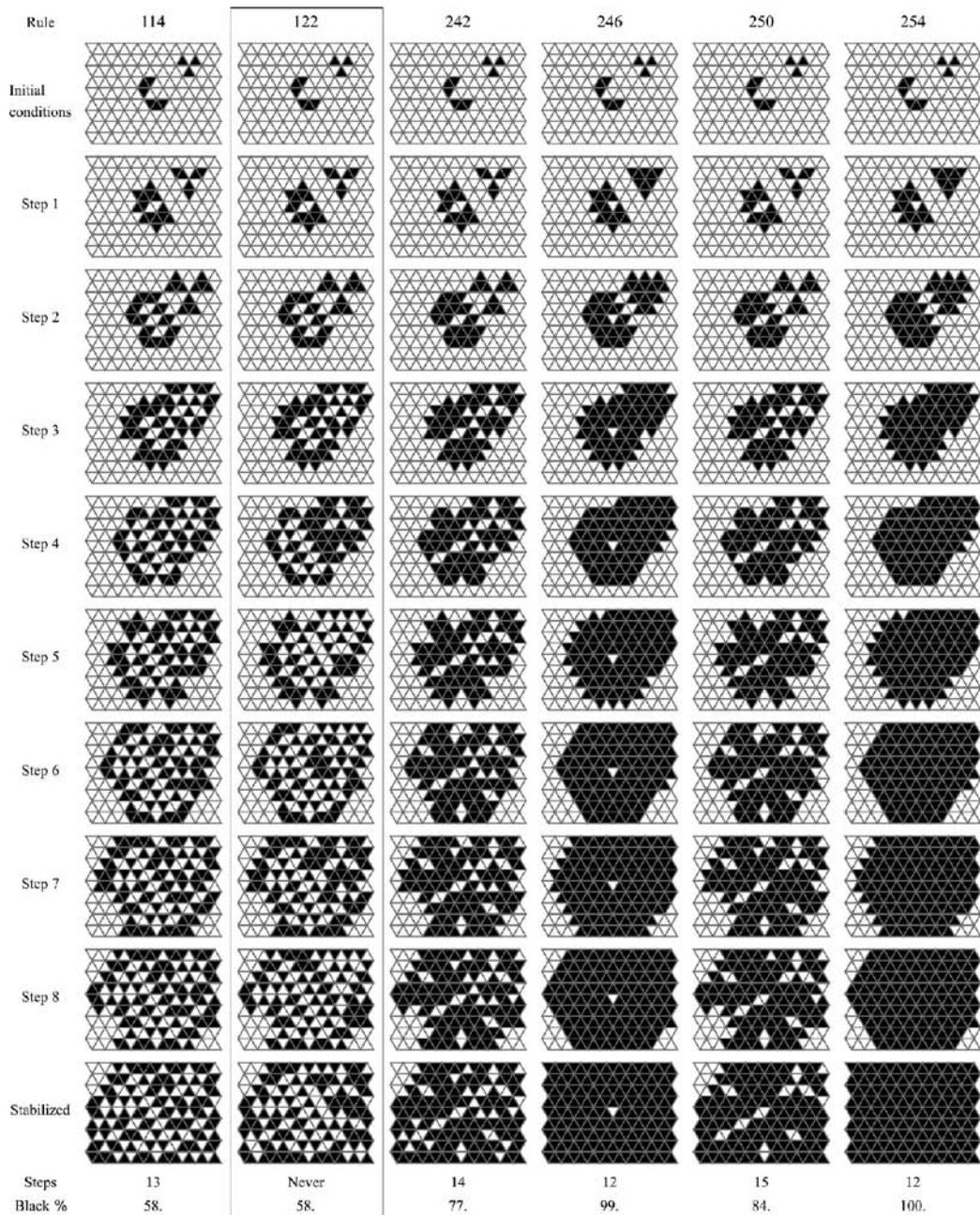


Figure 11: A few steps of evolution of the selected rules. On the bottom, the second last row shows the number of steps after which the pattern does not change; the last row indicates the number of black cells after stabilization. Rule 122 – the 25th step of evolution.

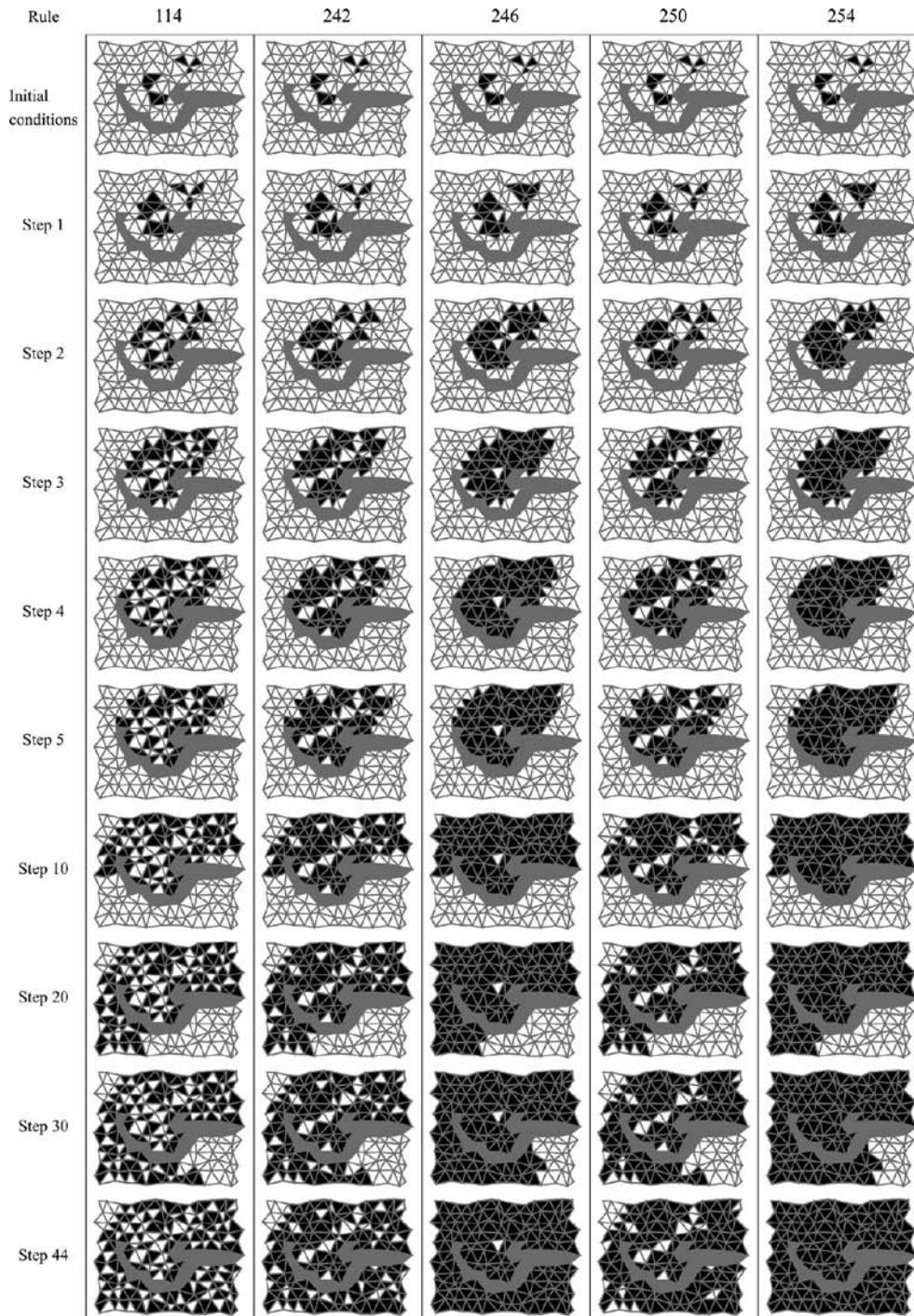


Figure 12: The evolution of all the appropriate TCA covering entirely the given irregular grid. The hole area is shown in gray.

rule 250 was selected because it stabilizes at high density with approximately 16% of visually significant artifacts.

8 THE HOLES

The notion of a ‘hole’ was introduced by using a new value – ‘2’ for the 2-color (0, 1) CA. This does not mean that it becomes a 3-color CA. The additional color and adequately modified rules prevent the CA from interacting with the area of a hole. The same procedure will be applied at the edge of the surface where cells have incomplete neighborhood, which is two or one neighbor. This method requires a certain modification of the TCA definition. Firstly, the ‘totalistic’ way of adding up the input from the neighbor cells was replaced by a ‘quasi-general’ way, as shown in Table 7 (see Appendix). The hole area value ‘2’ was introduced to the explicit set of rules as shown in Table 8 (see Appendix).

All five appropriate rules were applied on an irregular triangular grid with holes, as shown in Fig. 12.

9 APPLICATION ON A GIVEN 3D SURFACE

To demonstrate the concept on a larger triangulated 3D mesh a model of a duck was imported as shown in Fig. 13.

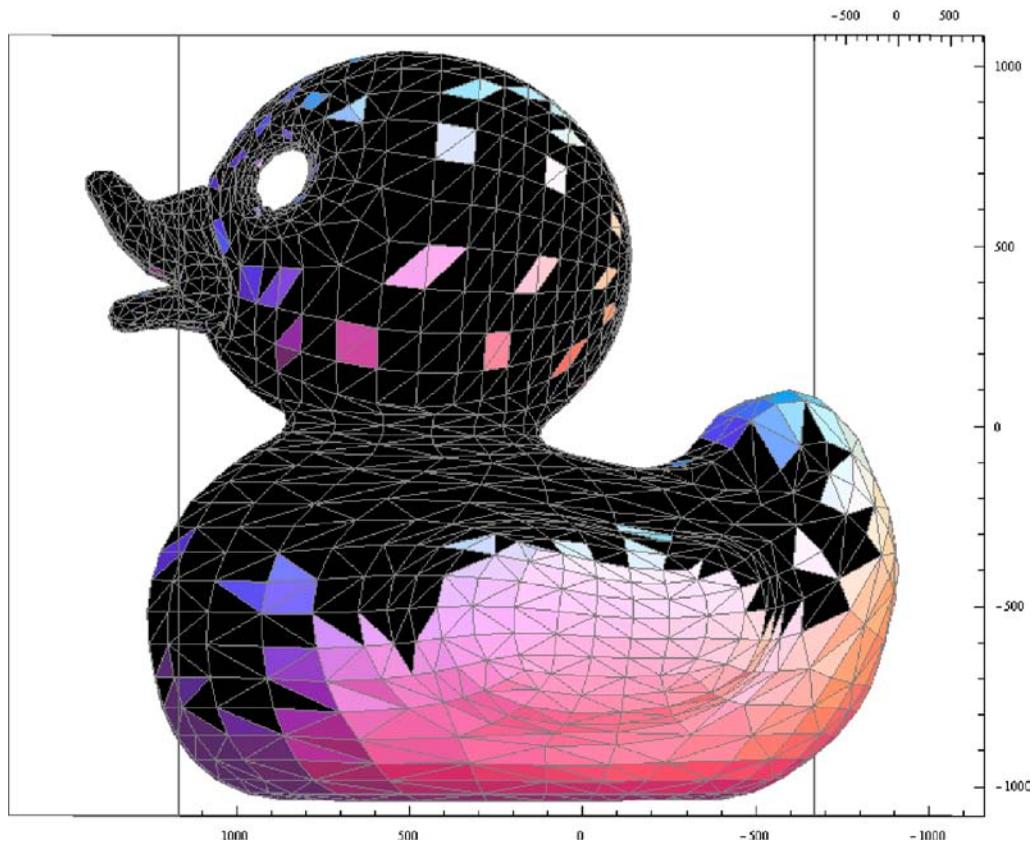


Figure 13: The imported 3D model shown with the #250 TCA propagating on the surface.

This choice can be justified for the following four reasons:

1. The model contains complicated 3D surfaces including holes.
2. It was available.
3. The shape is easily recognizable even at very different patterns on the surface.
4. The number of faces is optimal for the demonstration (approximately 4000).

The list of the triangular faces of the imported model was reordered according to the neighborhood. Initially all the neighbors of the first polygon were appended, followed by their neighbors and so forth. After that, all the faces were oriented in space so all of them had normal facing 'outwards'. Since the model contains 3976 faces, the manual setting of the initial conditions was difficult. For this purpose, the edge cells around the duck's eyes were used, since it is easy to detect the triangle on the edge (less than three neighbors).

At first, all the triangles sharing less than three pairs of vertices with any other three triangles were selected, assigned 2s as edge cells and moved to a temporary list.

Then the list of triangles was searched in the same way and this time assigned 1s as the initial conditions.

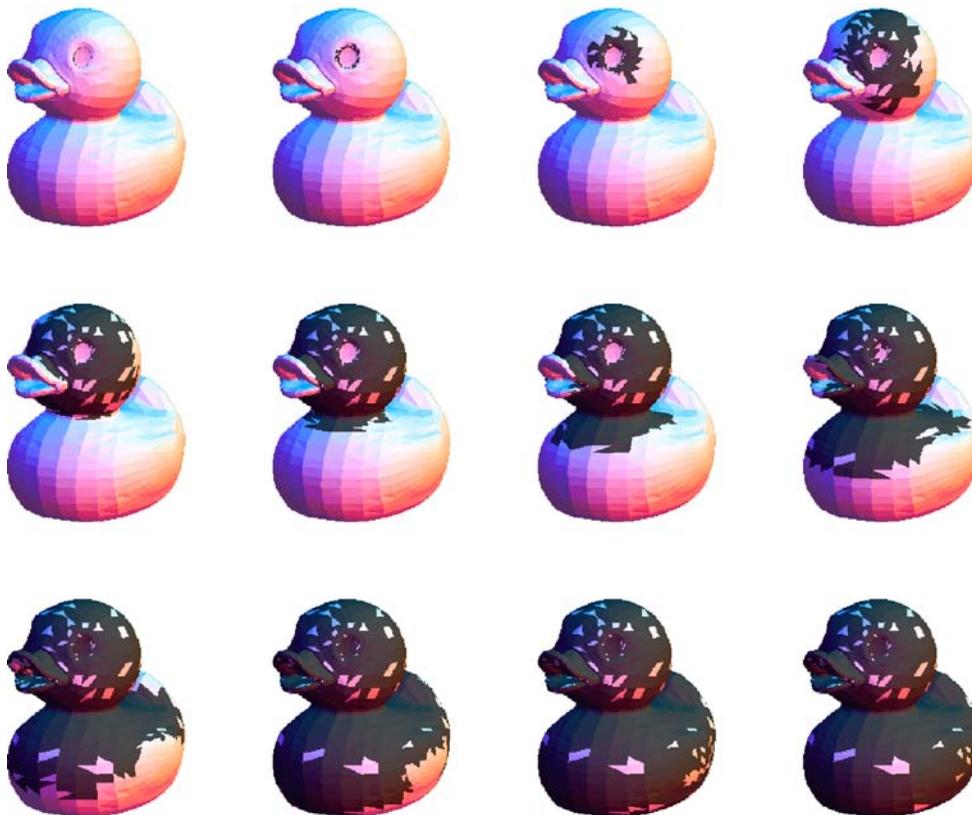


Figure 14: The #250 TCA applied on an imported 3D triangulated surface. The initial conditions and every 6th step of the evolution are shown; the pattern stabilizes after 78 steps.

Finally, both lists were joined to make a single list that contained both: triangles on the outer edge of the duck's eyes (with 2s as holes) and cells with the initial conditions on the second outermost ring around duck's eyes (assigned with 1s). The cells with 2s are not displayed (since they are holes!).

The process of running the CA on this 3D mesh is quite time consuming – a single step takes approximately 3.6 min on a single core 1.86 GHz computer. Computing nearly 80 steps took almost the whole night. The model with #250 TCA is shown in Fig. 14.

10 CONCLUSIONS

1. Semi-totalistic 2-color 2-dimension range-1 triangular cellular automata (ST2C2DR1TCA) were implemented successfully and can be applied on any 3D surface (after triangulation).
2. The pattern generated by certain rules spreads from the initial condition cells over the entire surface in an organic way.
3. The state of such a surface can be controlled to a certain extent by choosing the appropriate TCA rule and a set of initial conditions.
4. ST2C2DR1TCA allow the control of the state of the surface so that after the completion of the CA evolution 58, 77, 84, 99 or 100% of the cells are black.
5. Although in the computer simulation, the computation time of application of a TCA on a given surface depends on the complexity of the surface (number of triangles) it is not a major concern.

11 FUTURE WORK

1. A study on the general triangular 2DR1CA (the number of such rules is $2^{16} = 65536$). Totalistic and semi-totalistic rules are isotropic – the pattern develops uniformly in all directions. General rules can create directed patterns, which can be very practical, since it gives more control over the propagation of a pattern over the given surface.
2. The method of selecting the polygons for initial conditions and holes should be simplified, for example, by using colors in the list of polygons.
3. Optimization of the code for performing the CA computations more efficiently.
4. Development of a Graphical User Interface (GUI) to allow initial conditions on a 3D surface to be set up interactively.
5. Develop a better integration with a CAD system.
6. Use of a different type of neighborhood (e.g. Moore's) and higher ranges.
7. Carry out a study on how to control the whole 3D surface by controlling the initial conditions localized in certain areas such as at one small given area or around an edge and so on. Information from such studies could have significant practical applications.

ACKNOWLEDGMENTS

This project started at the NKS Summer School 2009 in Pisa, Italy, and was organized by Wolfram Research Inc.

I would like to thank Jason Cawley and Todd Rowland for their guidance, help and patience during the project.

The 3D triangulated model was imported from the MeshLab, a tool developed with the support of the 3D-CoForm project <http://meshlab.sourceforge.net/>.

REFERENCES

- [1] Zawidzki M., Implementing cellular automata for dynamically shading a building facade, *Complex Systems*, **3(18)**, 2009.
- [2] Zawidzki M. & Fujieda I., The prototyping of a shading device controlled by a cellular automaton, *Complex Systems*, **2(19)**, 2010.

- [3] Proksch E., Brandner J.M. & Jensen J.M., The skin: an indispensable barrier. *Exp Dermatol*, **17(12)**, pp. 1063–72, 2008. doi:10.1111/j.1600-0625.2008.00786.x
- [4] Madison K.C., Barrier function of the skin: “a raison d’être” of the epidermis. *J Invest Dermatol*, **121(2)**: pp. 231–41, 2003. doi:10.1046/j.1523-1747.2003.12359.x
- [5] Torcellini P. & Pless S., Trombe walls in low-energy buildings: practical experiences. *World Renewable Energy Congress VIII*, Denver, 2004.
- [6] Connor S., *The Book of Skin*, Cornell University Press, p. 176, 2003.
- [7] Wolfram S., *A New Kind of Science*, Wolfram Media, Inc., 2002.
- [8] Gans D. & Kuz Z., *The Organic Approach to Architecture*, Academy Press, 2003.
- [9] Bays C., Cellular automata in the triangular tessellation, *Complex Systems*, **8**, pp. 127–150, 1994.
- [10] Imai K., A computation-universal two-dimensional 8 state triangular reversible cellular automaton, *Theoretical Computer Science*, **231 (2)**, pp. 181–191, 2000. doi:10.1016/S0304-3975(99)00099-7
- [11] Alonso-Sanz, R., A structurally dynamic cellular automaton with memory in the triangular tessellation, *Complex Systems*, **17(1)**, pp. 1–15, 2007.
- [12] Mitchell M., Hraber P.T. & Crutchfield J.P., Revisiting the edge of chaos: evolving cellular automata to perform computations, *Complex Systems*, **7**, pp. 89–130, 1993.

APPENDIX

Table 6: Thirty-two ‘promising’ TCA rules, from a total of 256, were chosen for further investigation.

18	0	1	0	0	1	0	0	0
22	0	1	1	0	1	0	0	0
26	0	1	0	1	1	0	0	0
30	0	1	1	1	1	0	0	0
50	0	1	0	0	1	1	0	0
54	0	1	1	0	1	1	0	0
58	0	1	0	1	1	1	0	0
62	0	1	1	1	1	1	0	0
82	0	1	0	0	1	0	1	0
86	0	1	1	0	1	0	1	0
90	0	1	0	1	1	0	1	0
94	0	1	1	1	1	0	1	0
114	0	1	0	0	1	1	1	0
118	0	1	1	0	1	1	1	0
122	0	1	0	1	1	1	1	0
126	0	1	1	1	1	1	1	0
146	0	1	0	0	1	0	0	1
150	0	1	1	0	1	0	0	1
154	0	1	0	1	1	0	0	1
158	0	1	1	1	1	0	0	1
178	0	1	0	0	1	1	0	1
182	0	1	1	0	1	1	0	1
186	0	1	0	1	1	1	0	1

Continued

Table 6: *Continued*

190	0	1	1	1	1	1	0	1
210	0	1	0	0	1	0	1	1
214	0	1	1	0	1	0	1	1
218	0	1	0	1	1	0	1	1
222	0	1	1	1	1	0	1	1
242	0	1	0	0	1	1	1	1
246	0	1	1	0	1	1	1	1
250	0	1	0	1	1	1	1	1
254	0	1	1	1	1	1	1	1

Each rule has values preset at three positions indicated by a gray background. Positions are counted from the right. Position 4: $\{1,0\} \rightarrow 1$ and 7: $\{0,1\} \rightarrow 1$ ensure that a single black cell without any black neighbors will not ‘disappear’ and position 8: $\{0,0\} \rightarrow 0$ ensures that the pattern will not turn instantly from white to black, in other words that a black cell cannot be produced from ‘nothing’. The first column shows the decimal rule number encoding of the binary encoding.

Table 7: The semi-totalistic rules are unfolded to explicit rules. Example: #114 TCA.

$\{1,3\} \rightarrow 0$	$\{1,1,1,1\} \rightarrow 0$
$\{1,2\} \rightarrow 1$	$\{1,1,1,0\} \rightarrow 1, \{1,1,0,1\} \rightarrow 1, \{1,0,1,1\} \rightarrow 1$
$\{1,1\} \rightarrow 1$	$\{1,1,0,0\} \rightarrow 1, \{1,0,1,0\} \rightarrow 1, \{1,0,0,1\} \rightarrow 1$
$\{1,0\} \rightarrow 1$	$\{1,0,0,0\} \rightarrow 1$
$\{0,3\} \rightarrow 0$	$\{0,1,1,1\} \rightarrow 0$
$\{0,2\} \rightarrow 0$	$\{0,1,1,0\} \rightarrow 0, \{0,1,0,1\} \rightarrow 0, \{0,0,1,1\} \rightarrow 0$
$\{0,1\} \rightarrow 1$	$\{0,1,0,0\} \rightarrow 1, \{0,0,1,0\} \rightarrow 1, \{0,0,0,1\} \rightarrow 1$
$\{0,0\} \rightarrow 0$	$\{0,0,0,0\} \rightarrow 0$

Table 8: Eight-one explicit rules of the #114 TCA prepared for operating on a mesh with holes.

$\{1,1,1,1\} \rightarrow 0, \{1,1,1,0\} \rightarrow 1, \{1,1,0,1\} \rightarrow 1, \{1,0,1,1\} \rightarrow 1, \{1,1,0,0\} \rightarrow 1,$ $\{1,0,1,0\} \rightarrow 1, \{1,0,0,1\} \rightarrow 1,$ $\{1,0,0,0\} \rightarrow 1, \{0,1,1,1\} \rightarrow 0, \{0,1,1,0\} \rightarrow 0, \{0,1,0,1\} \rightarrow 0, \{0,0,1,1\} \rightarrow 0,$ $\{0,1,0,0\} \rightarrow 1, \{0,0,1,0\} \rightarrow 1,$ $\{0,0,0,1\} \rightarrow 1, \{0,0,0,0\} \rightarrow 0, \{1,2,2,2\} \rightarrow 1, \{1,2,2,1\} \rightarrow 1, \{1,2,2,0\} \rightarrow 1,$ $\{1,2,1,2\} \rightarrow 1, \{1,2,1,1\} \rightarrow 1,$ $\{1,2,1,0\} \rightarrow 1, \{1,2,0,2\} \rightarrow 1, \{1,2,0,1\} \rightarrow 1, \{1,2,0,0\} \rightarrow 1, \{1,1,2,2\} \rightarrow 1,$ $\{1,1,2,1\} \rightarrow 1, \{1,1,2,0\} \rightarrow 1,$ $\{1,1,1,2\} \rightarrow 1, \{1,1,0,2\} \rightarrow 1, \{1,0,2,2\} \rightarrow 1, \{1,0,2,1\} \rightarrow 1, \{1,0,2,0\} \rightarrow 1,$ $\{1,0,1,2\} \rightarrow 1, \{1,0,0,2\} \rightarrow 1,$ $\{0,2,2,2\} \rightarrow 0, \{0,2,2,1\} \rightarrow 1, \{0,2,2,0\} \rightarrow 0, \{0,2,1,2\} \rightarrow 1, \{0,2,1,1\} \rightarrow 0,$ $\{0,2,1,0\} \rightarrow 1, \{0,2,0,2\} \rightarrow 0,$	<p>The original rules converted to quasi general rules.</p> <p>2 acts like a 0</p>
--	--

Continued

Table 8: *Continued*

$\{0,2,0,1\} \rightarrow 1, \{0,2,0,0\} \rightarrow 0, \{0,1,2,2\} \rightarrow 1, \{0,1,2,1\} \rightarrow 0, \{0,1,2,0\} \rightarrow 1,$	
$\{0,1,1,2\} \rightarrow 0, \{0,1,0,2\} \rightarrow 1,$	
$\{0,0,2,2\} \rightarrow 0, \{0,0,2,1\} \rightarrow 1, \{0,0,2,0\} \rightarrow 0, \{0,0,1,2\} \rightarrow 1, \{0,0,0,2\} \rightarrow 0,$	
$\{2,1,1,1\} \rightarrow 2, \{2,1,1,0\} \rightarrow 2, \{2,1,0,1\} \rightarrow 2, \{2,1,0,0\} \rightarrow 2, \{2,0,1,1\} \rightarrow 2,$	2 never changes
$\{2,0,1,0\} \rightarrow 2, \{2,0,0,1\} \rightarrow 2,$	
$\{2,0,0,0\} \rightarrow 2, \{2,0,0,2\} \rightarrow 2, \{2,0,1,2\} \rightarrow 2, \{2,0,2,0\} \rightarrow 2, \{2,0,2,1\} \rightarrow 2,$	
$\{2,0,2,2\} \rightarrow 2, \{2,1,0,2\} \rightarrow 2,$	
$\{2,1,1,2\} \rightarrow 2, \{2,1,2,0\} \rightarrow 2, \{2,1,2,1\} \rightarrow 2, \{2,1,2,2\} \rightarrow 2, \{2,2,0,0\} \rightarrow 2,$	
$\{2,2,0,1\} \rightarrow 2, \{2,2,0,2\} \rightarrow 2,$	
$\{2,2,1,0\} \rightarrow 2, \{2,2,1,1\} \rightarrow 2, \{2,2,1,2\} \rightarrow 2, \{2,2,2,0\} \rightarrow 2, \{2,2,2,1\} \rightarrow 2,$	
$\{2,2,2,2\} \rightarrow 2$	
