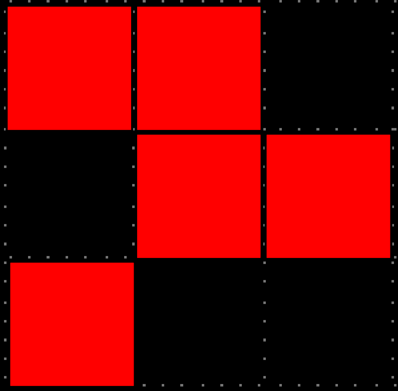


Cellular Automaton Systems

Alexandra Stamou

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

caad DARCH

Cellular automata:

an idealization of a physical system in which space and time are discrete, and the physical quantities take only a finite set of values

simple mathematical models of computation which exhibit fascinatingly complex behaviour

self-organization

Cellular Automata, often referred to as CA, are an idealization of a physical system, in which space and time are discrete, and the physical quantities take only a finite set of values. They are simple mathematical models of computation which exhibit fascinatingly complex behaviour, and for this reason they are used to investigate self-organization.

According to the description given by Stephen Wolfram, a cellular automaton consists of a regular uniform lattice (or “array”), usually infinite in extend, with a discrete variable at each site (“cell”). The state of a cellular automaton is completely specified by the values of variables at each site. A cellular automaton evolves in discrete time steps, with the value of a variable at one site (t+1) being affected by the values of variables at sites in its “neighbourhood” on the previous time step (t).[A7] The variables at each site are updated simultaneously, based on the values of the variables in their neighbourhood at the preceding time step, and according to a definite set of “local” rules.

Formally we can restate the principles of CA as embodying four distinct characteristics: first there are the cells, objects in any dimensional space but with an adjacency to one another. Second there is the state of the cell. Each cell can take only one state at any one time from a set of states. Third there is the neighborhood of a cell, the neighborhood being the immediately adjacent set of cells that are “next” to the cell. Finally, there are transition rules that drive the changes of the state in a cell, as a function that results from what exists or is happening in the cell’s neighborhood.

It is assumed that the transition rules must be uniform, which means they must apply to each cell, state and neighborhood at all times; and that every change in state must be local. There are also initial and boundary conditions, which specify the start and end points of the simulation in space and time.[A19]

These dynamical systems, which are fully discrete in space and time, operate on a uniform regular lattice and are characterized by “local” interactions, are the so-called cellular automata.[B23] a cellular automaton consists of a regular uniform lattice, usually infinite in extend, with a discrete variable at each site

The state of a cellular automaton is completely specified by the values of variables at each site

A cellular automaton evolves in discrete time steps, with the value of a variable at one site (t+1) being affected by the values of variables at sites in its “neighbourhood” on the previous time step (t)

principles of cellular automata

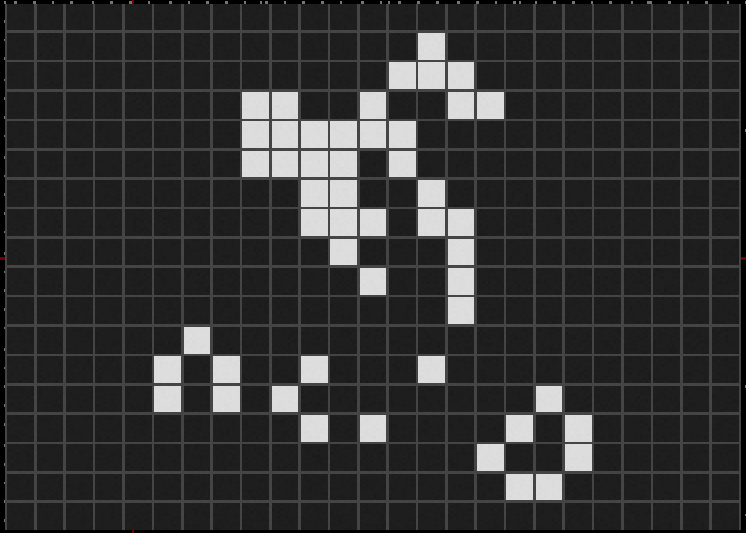
four distinct characteristics :

1 cells

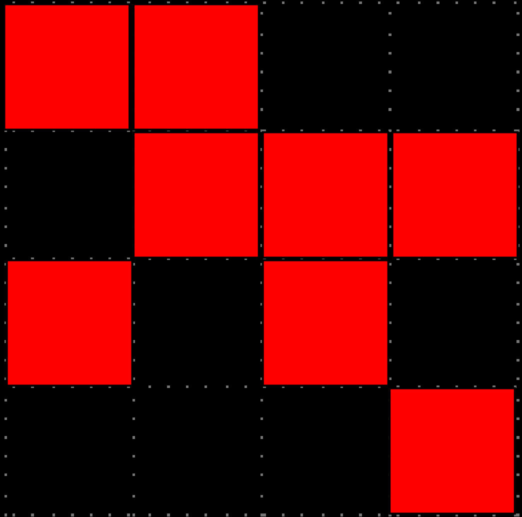
2 state of the cell

3 neighborhood of a cell

4 transition rules



Alexandra Stamou



A Brief History of Cellular Automata

Alexandra Stamou

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

caad DARCH

John von Neumann

The concept of a cellular automaton dates back from the late 1940s. The pioneer is John von Neumann, a Hungarian mathematician, who, at the end of the 1940s, was involved in the design of the first digital computers. His concept of cellular automata constitutes the first applicable model of massively parallel computation.A8 Parallel computation is the simultaneous execution of the same task on multiple processors in order to obtain results faster. The idea is based on the fact that a process of solving a problem usually can be divided into smaller discrete tasks, which may be carried out simultaneously with some coordination.B1 Von Neumann was interested in building a machine which could have the same complexity as that of a human brain. He thought that such a machine should also contain self-control and self-repair mechanisms. Through this concept he decided to define the properties a machine should have in order to be self-replicating.

With the help of Stanislaw Marcin Ulam, a mathematician and scientist, who studied the growth of crystals in the 1940s using a simple lattice network, von Neumann focused on finding a logical abstraction of the self-reproduction mechanism, and began working in a framework of a fully discrete universe made up of cells. Each cell was characterized by an internal state, which typically consisted of a finite number of information bits. He then suggested that this system of cells evolves, in discrete time steps, and this process is determined by a rule, which is the same for each cell and is a function of the states of the neighbor cells. Actually, the state of a cell at a time slice (t+1) is a function of the state of a finite number of neighbor cells at the previous time slice (t). Moreover the up-dating of the status of each cell is happening synchronously.

What is quite interesting is that the cells of these systems bear in themselves the recipe to generate new identical cells. Although one would not expect that a machine can built an object of the same complexity as itself, with cellular automata one obtains a system, which is able to generate new systems of the same level of complexity and capabilities.A8

Von Neumann was interested in building a machine which could have the same complexity as that of a human brain.

He thought that such a machine should also contain self-control and self-repair mechanisms.

Through this concept he decided to define the properties a machine should have in order to be self-replicating

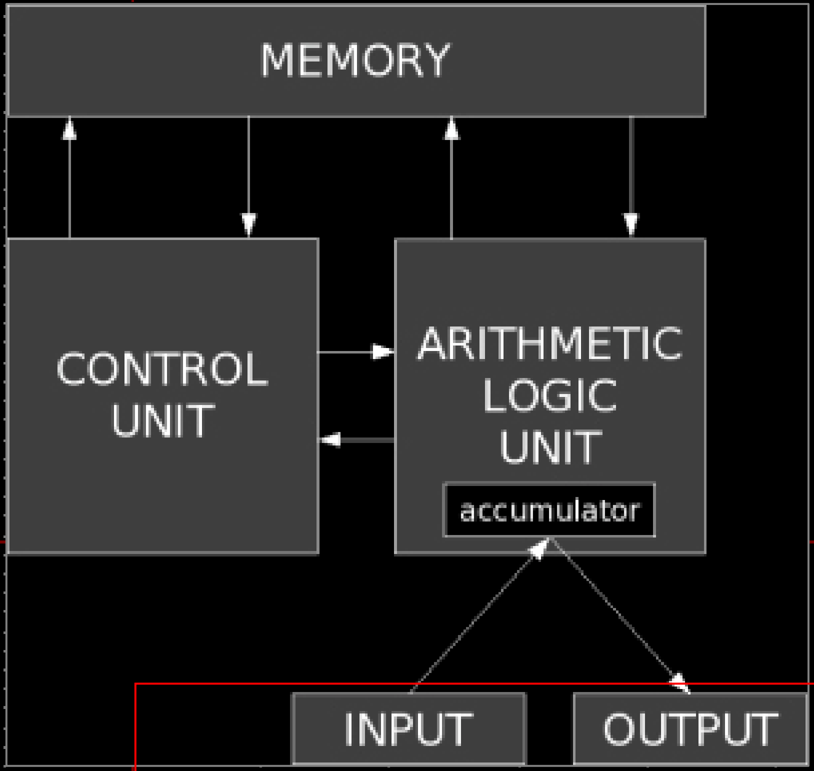
Stanislaw Marcin Ulam

The von Neumann cellular automaton is the first self-replicating CA, which is composed of a two-dimensional square lattice and of several thousand elementary cells with the ability to self-reproduce. Each one of the cells has up to 29 possible states. The rule that controls the evolution requires the state of each cell, and its four nearest neighbors located north, east, south and west. The von Neumann rule has the so-called property of universal computation, which means that the initial configuration of the cellular automaton can lead to a solution of any computer algorithm.A8 More specifically, given the description of any machine, the universal constructor will locate the proper parts, and will construct the machine. Furthermore it will contain a description copier, which will in fact attach a copy of the description of the machine to the offspring machine.A13 But this property is more of a theoretical interest than of a practical one. Its conclusion suggests that a cellular automaton with a simple rule can actually exhibit a complex and unpredictable behavior.

After the work of von Neumann, many other scientists worked on simpler cellular automata, one of which was C.G.Langton, who studied a cellular automaton with only eight states, able to self-replicate. This simplification was made possible by giving up the property of universal computation, and introducing a spatial distributed sequence of instructions, which is executed to create a new structure and then is entirely copied in this new structure.A8

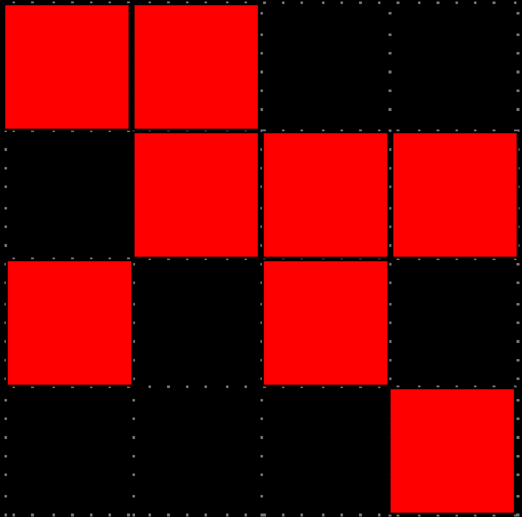
Von Neumann cellular automaton

The von Neumann cellular automaton is the first self-replicating CA, which is composed of a two-dimensional square lattice and of several thousand elementary cells with the ability to self-reproduce. Each one of the cells has up to 29 possible states. The rule that controls the evolution requires the state of each cell, and its four nearest neighbors located north, east, south and west.



The von Neumann rule has the so-called property of universal computation, which means that the initial configuration of the cellular automaton can lead to a solution of any computer algorithm

Von Neumann cellular universal machine

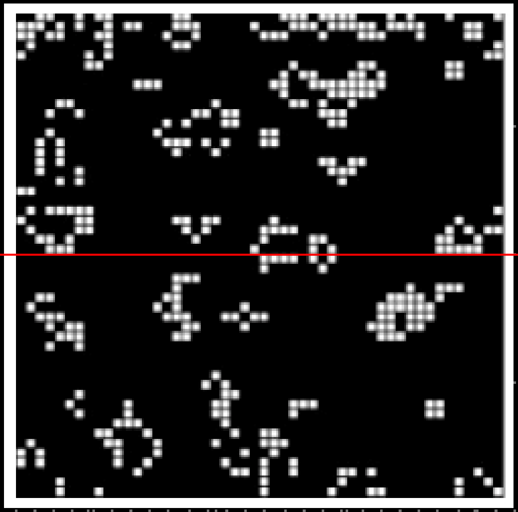


Simple Dynamical Systems and the “Game of Life”

Alexandra Stamou

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

caad DARCH



John Horton Conway

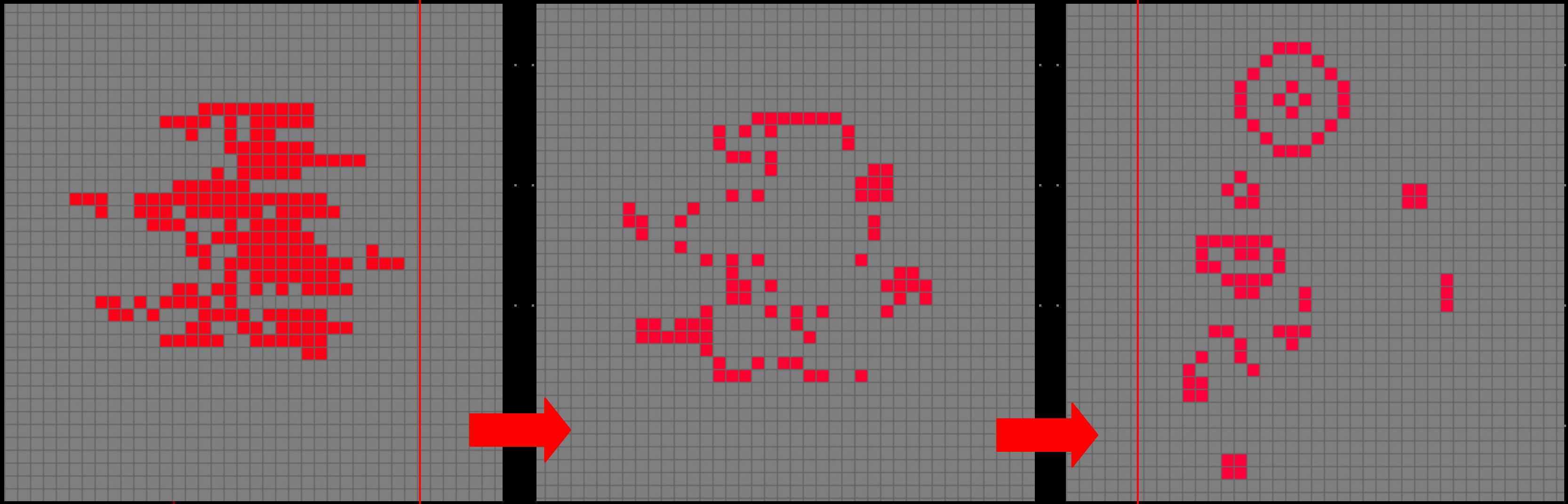
“Life” combined all the notions of cellular automata into a model that simulated the key elements of reproduction in the simplest possible way

In 1970 the concept of the cellular automata was brought to the attention of a wide audience through the introduction of a simple ecological model, called “Game of Life”, by the British mathematician John Horton Conway. Martin Gardner wrote an article in the October 1970 issue of Scientific American, titled: “Mathematical Games: The fantastic combinations of John Conway’s new solitaire game “life”. “Life” combined all the notions of cellular automata into a model that simulated the key elements of reproduction in the simplest possible way. Its popularity rests mainly upon the fact that a generation of hackers took up Conway’s idea and explored in countless ways the kinds of complexity that emerge from such simplicity.[A19]

Conway was interested in the self-replication logic of machines and he attempted to simplify von Neumann’s ideas. His motivation was to find a simple rule leading to complex behaviours. He imagined an infinite two-dimensional square lattice, like a checkerboard, in which each cell can be in one of two possible states: dead (state 0) or alive (state 1). The updating rule of the Game of Life is as follows: a dead cell surrounded by exactly three living cells comes back to life. On the other hand, a living cell surrounded by less than two or more than three neighbours dies, as if by loneliness or overcrowding respectively. In the case of the Game of Life, each cell is affected by the state of its eight neighbours, which are the cells that are directly horizontally, vertically, or diagonally adjacent. [A8, B7]

This "game" is actually a zero-player game, meaning that its evolution is determined by its initial state, needing no input from human players. One interacts with the Game of Life by creating an initial configuration and observing how it evolves.[B7]

It turned out that the game of life automaton has an unexpectedly rich behaviour. Many different patterns and complex structures emerge out of simple initial configurations, including static patterns (“still lifes”), repeating patterns, which are patterns that return to their original state, after a finite number of generations (“oscillators”), and patterns that translate themselves across the board (“spaceships”), such as the “gliders”, which correspond to a particular arrangement of adjacent cells that has the property to move across space, along straight trajectories. [A8, B7]

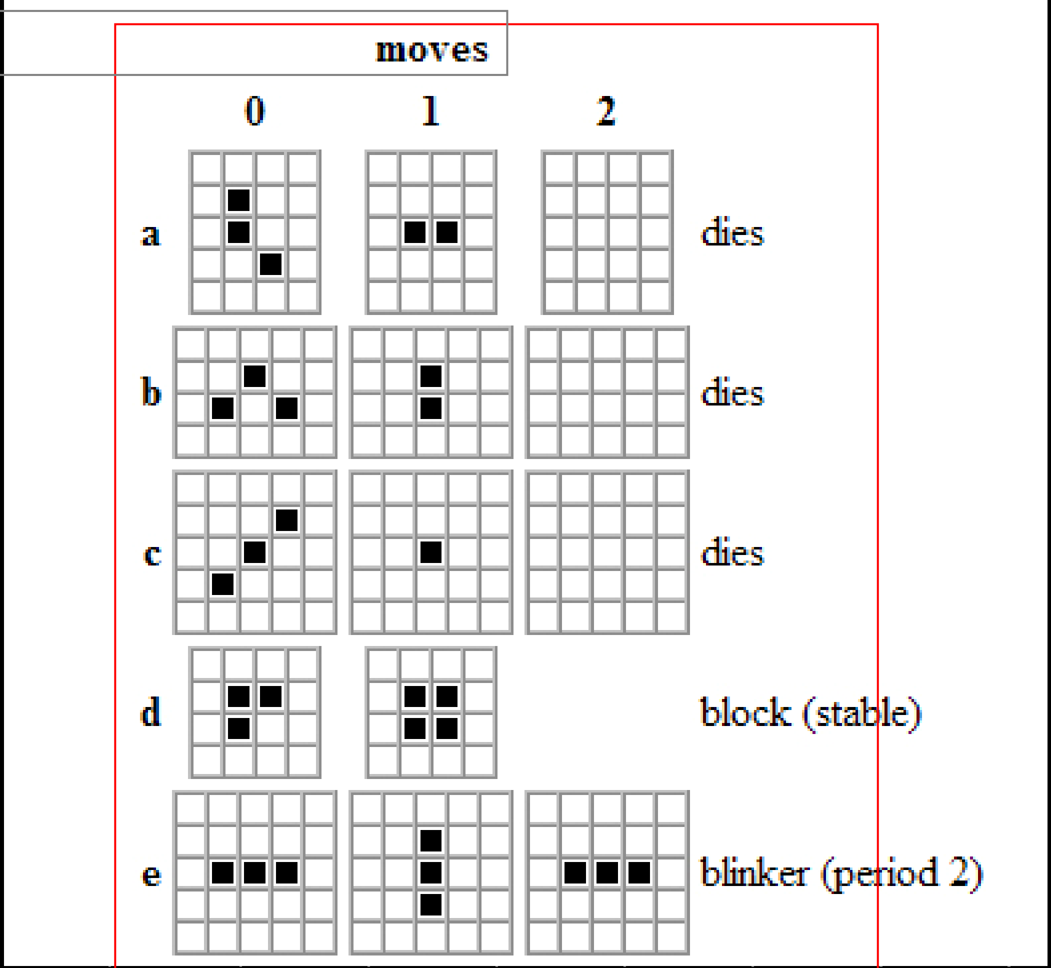


Conway imagined an infinite two-dimensional square lattice, like a checkerboard, in which each cell can be in one of two possible states: dead (state 0) or alive (state 1).

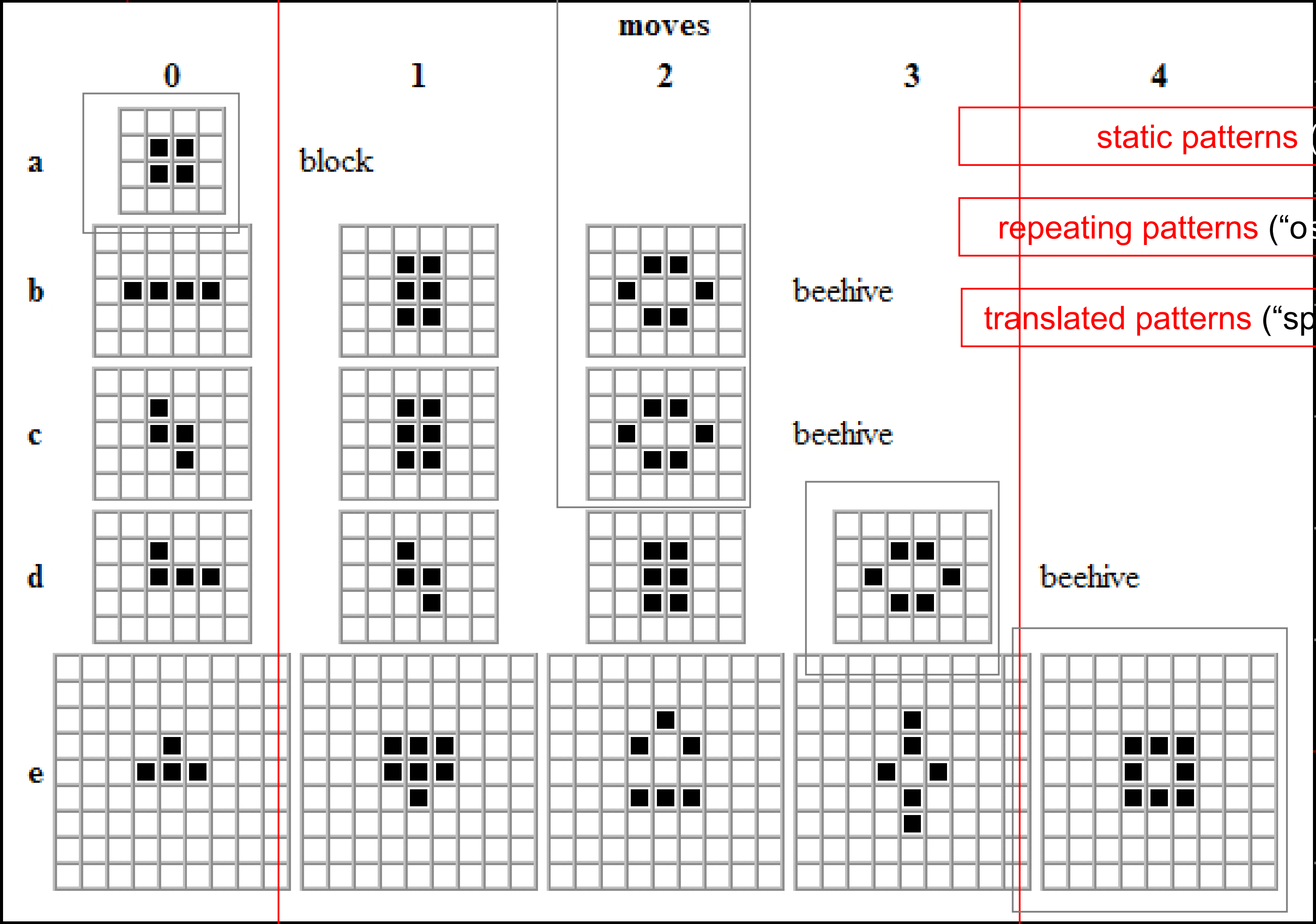
“I have often made the hypothesis that ultimately Physics will not require a mathematical statement, that in the end the machinery will be revealed, and the laws will turn out to be simple, like the checker board with all its apparent complexities.”

Richard Feynman

Nobel Prize in Physics, 1965



The updating rule of the Game of Life is as follows: a dead cell surrounded by exactly three living cells comes back to life. On the other hand, a living cell surrounded by less than two or more than three neighbours dies, as if by loneliness or overcrowding respectively. In the case of the Game of Life, each cell is affected by the state of its eight neighbours, which are the cells that are directly horizontally, vertically, or diagonally adjacent.

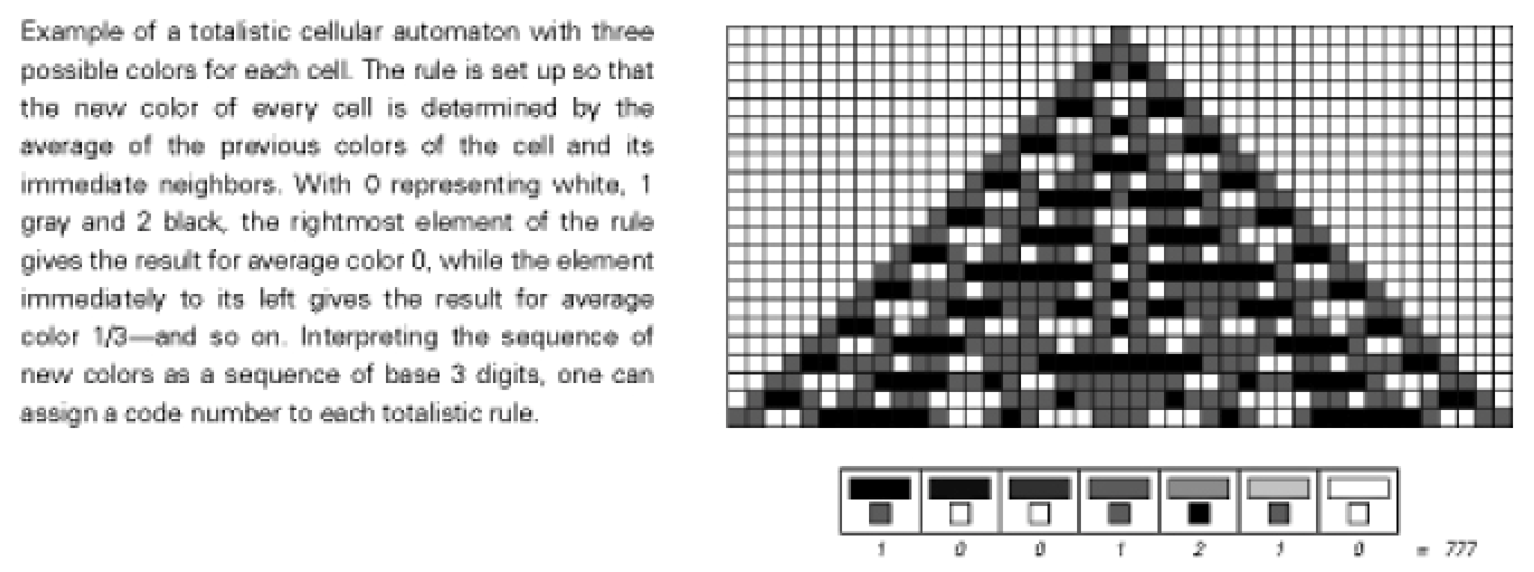


static patterns (“still lifes”)

repeating patterns (“oscillators ”)

translated patterns (“spaceships”)

different patterns and complex structures emerge out of simple initial configurations

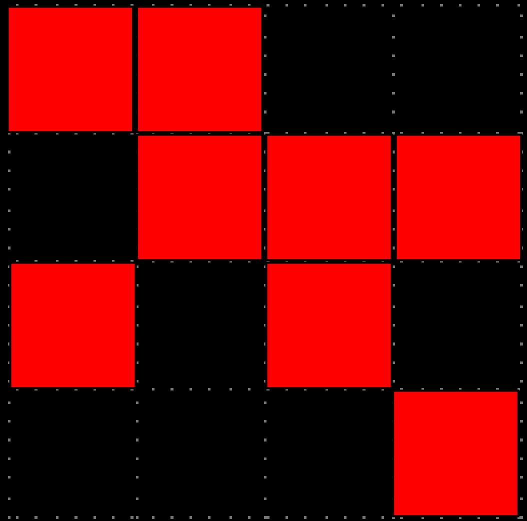


Example of a totalistic cellular automaton with three possible colors for each cell. The rule is set up so that the new color of every cell is determined by the average of the previous colors of the cell and its immediate neighbors. With 0 representing white, 1 gray and 2 black, the rightmost element of the rule gives the result for average color 0, while the element immediately to its left gives the result for average color 1/3—and so on. Interpreting the sequence of new colors as a sequence of base 3 digits, one can assign a code number to each totalistic rule.

Game of Life → Totalistic Cellular Automata

Game of Life → computational universality

The Game of Life is an example of a special class of the so-called “totalistic” cellular automata, in which the value of a site depends only on the sum of the values of its neighbours at the previous time step, and not on their individual values. As far as the von Neumann rule, the Game of Life automaton has been shown to have the important property of computational universality. The proof of this is the existence of cellular automaton structures which emulate components (such as “wires”) of a standard digital computer.A7



Analysis of Cellular Automata

Alexandra Stamou

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

caad DARCH

“Cellular automata are sufficiently simple to allow detailed mathematical analysis, but sufficiently complex to exhibit a wide variety of complicated phenomena”.

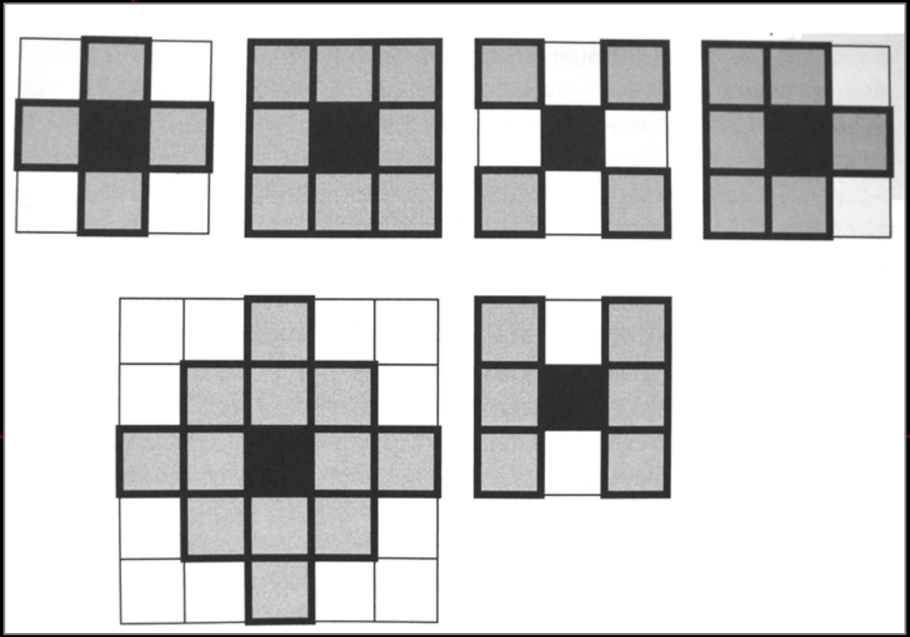
Stephen Wolfram, in Cellular Automata and Complexity, 1994

In the early 1980’s, Stephen Wolfram, a British scientist known for his work in theoretical particle physics, complexity theory and computer algebra, and also known as the creator of the computer program Mathematica, investigated the typical behavior of simple arbitrarily chosen programs, created without any specific task in mind. One class of these programs was the cellular automata. Wolfram studied in detail a family of simple one-dimensional cellular automata rules (the now famous Wolfram rules). He noticed that a cellular automaton is a discrete dynamical system that exhibits various complex behaviors, yet in a simple framework. After this observation, it was made obvious that, a concept such as complexity could be investigated on mathematical models allowing an exact numerical computer calculation, due to their Boolean nature (no numerical errors and no truncation as in more traditional models).[A8]

Before moving further into examining some examples of specific cellular automaton rules, it is important to refer to the different types of neighborhoods that define the number of neighboring cells, which may affect the state of one cell. The first neighborhood is called the Von Neumann Neighborhood, which is symmetrically arranged. The same stands for the second neighborhood, which is called Moore Neighborhood, as well as for the third, which is actually a Displaced von Neumann neighborhood. The Moore Neighborhood is the most usual and the most general. After the asymmetric, there is another neighborhood, which is a combination of the von Neumann and the Moore neighborhoods and it is called ”MvN”. Another example is the “H” neighborhood, which is a semi symmetric structure. [A19]

At this point it is important for one to understand the enormous array of possible structures that can be built from simple bits using the cellular automata approach. In general, with D states and K neighbors in each neighborhood, there are D_K different configurations of cells that can result, and when transition rules are considered, there are D_{DK} varieties of CA that can exist. This means that any conceivable pattern might be computed using CA. For example, in the case of the von Neumann neighborhood, which consists of five cells, there are 25 or 32 different configurations of on-off cell patterns that affect the transition rule.[A19]

Types of neighborhoods



six types :

1 Von Neumann

2 Moore Neighborhood

3 Displaced von Neumann

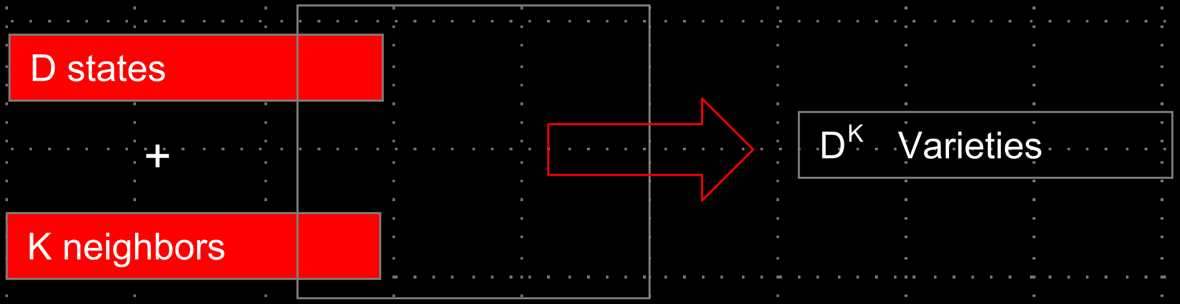
4 Asymmetric

5 MvN

6 H Neighborhood

An enormous array of possible structures

in each neighborhood:



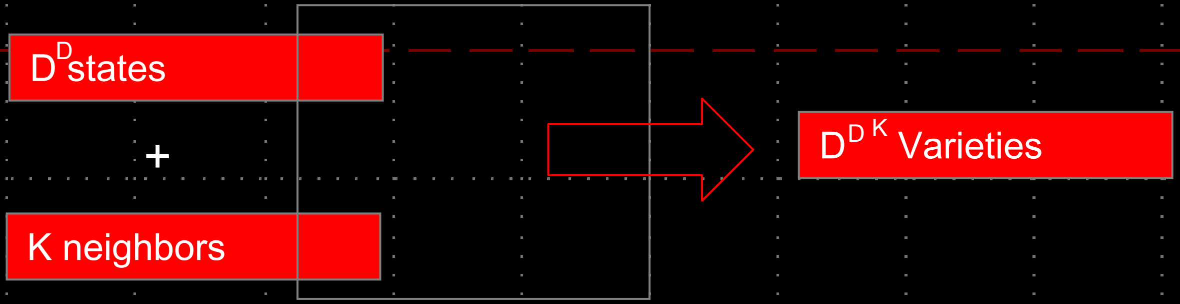
D states

+

K neighbors

D^K Varieties

with transition rules on the configurations:



D^D states

+

K neighbors

D^{D^K} Varieties

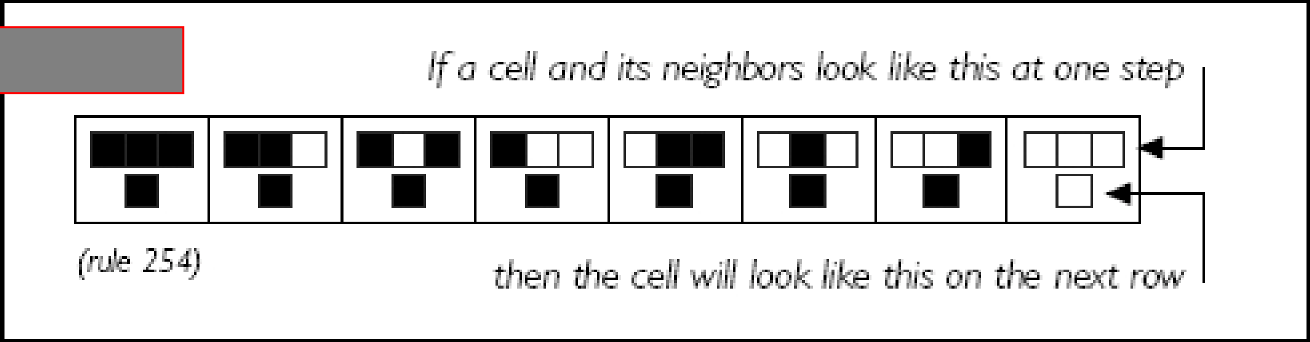
One of the rules investigated by Wolfram is cellular automaton rule 254. In his book "A New Kind Of Science", this rule, among others, is illustrated with a visual representation of the behavior of a cellular automaton, with each row of cells corresponding to one step, or generation. At the first step the cell in the center of the row is black, and all the other cells are white. At every sequent step(t+1) there is a definite rule that determines the color of a given cell from the color of that cell and its immediate left and right neighbors on the step before (t). According to this rule, on each successive step, a particular cell is made black whenever it or any of its neighbors were black on the step before. This leads to a simple expanded pattern uniformly filled with black.A6

But if this rule is slightly modified, then a different pattern will immediately emerge. For example, rule 250 makes a particular cell black if either of its neighbors was black on the step before, and makes a cell white if both of its neighbors were white. Starting from a single cell, this rule leads to a checkerboard-like pattern. Rule 90 constitutes a slightly modified version of the previous rule: in this case a cell is made black if either of its neighbors – but not both – was black on the step before. The pattern this rule produces is not any more so simple, as the previous ones. And if the cellular automaton is let to run for more steps, then a rather intricate pattern emerges. But a closer look will show that although intricate, it actually consists of many nested triangular pieces, with exactly the same form. Moreover each of these pieces is essentially a smaller copy of the whole pattern.A6

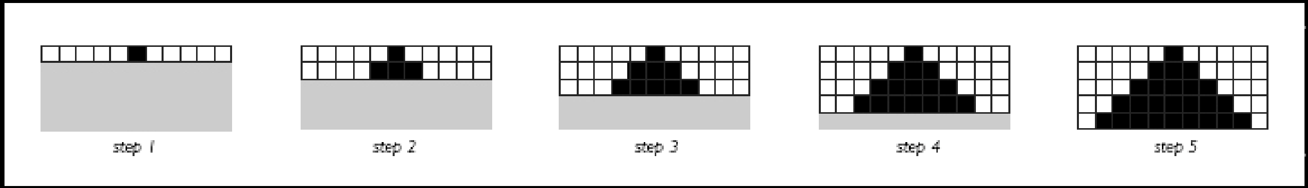
In all the above three examples the patterns that are produced are more or less regular: the first one is described as a simple uniform pattern, the second one as a repetitive, and the third one as an intricate, yet nested pattern. But there are other cellular automaton rules, which although of the same simplicity, they produce a far more complex and unpredictable pattern. One paradigm is rule 30. According to this rule the program has first to examine a given cell and its immediate right neighbor. If both were white on the previous step, then the new color of the given cell should get the same color with the previous color of its immediate left neighbor. Otherwise it should get the opposite color of that. The pattern that is produce by this rule seems in many respects random. It is of high complexity and shows almost no overall regularity. A6

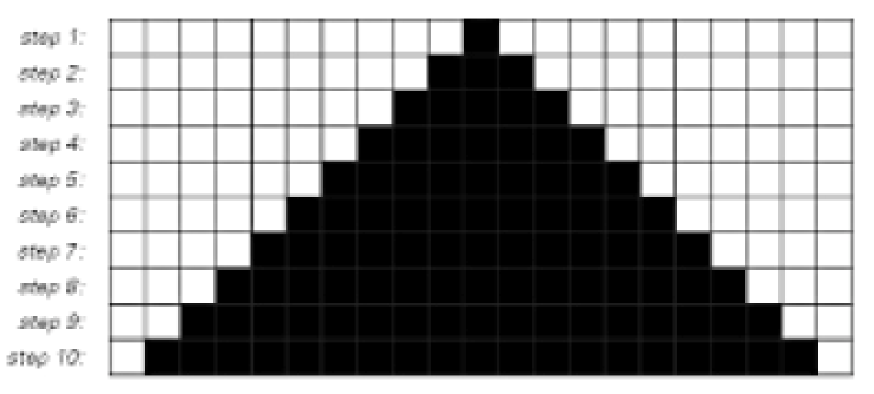
Some Wolfram Cellular Automaton Rules

rule 254



with each row of cells corresponding to one step, or generation



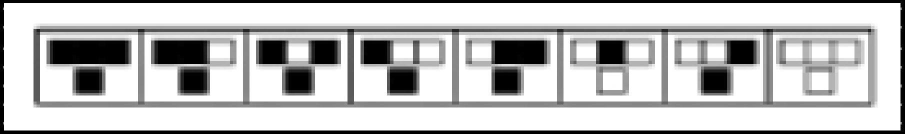


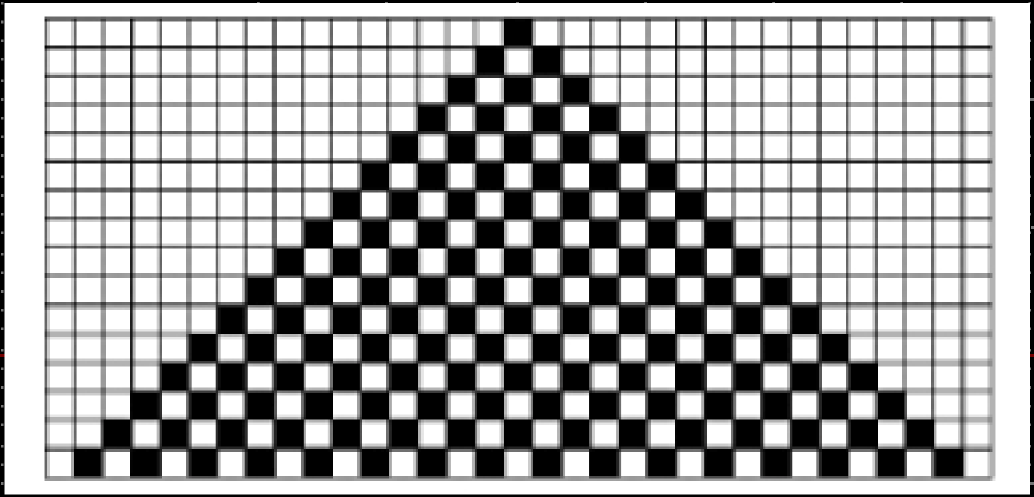
a particular cell is made black whenever it or any of its neighbors were black on the step before

simple pattern uniformly filled

Some Wolfram Cellular Automaton Rules

rule 250



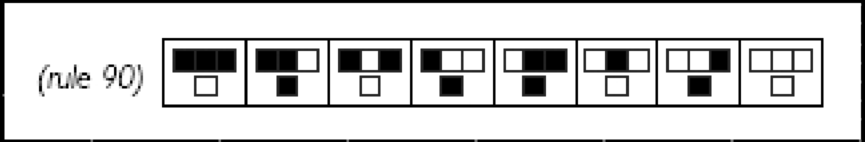


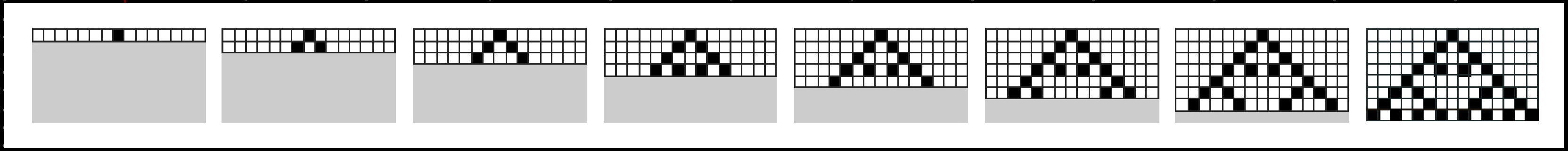
this rule makes a particular cell black if either of its neighbors was black on the step before, and makes a cell white if both of its neighbors were white

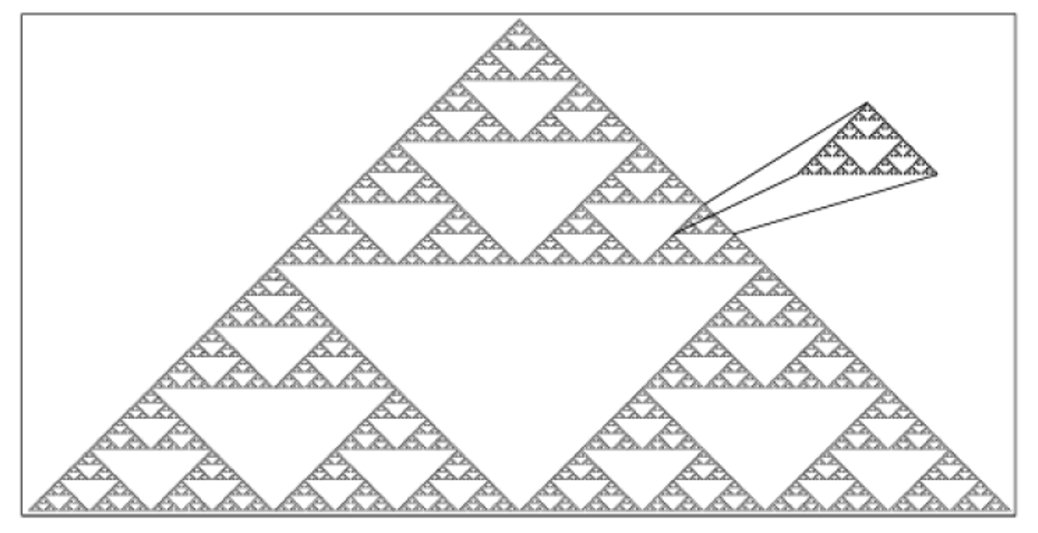
checkerboard-like pattern

Some Wolfram Cellular Automaton Rules

rule 90





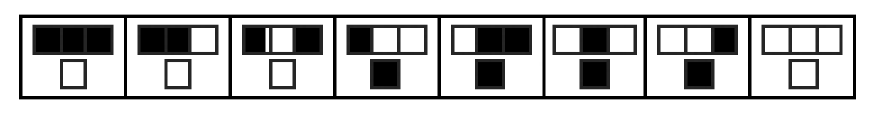


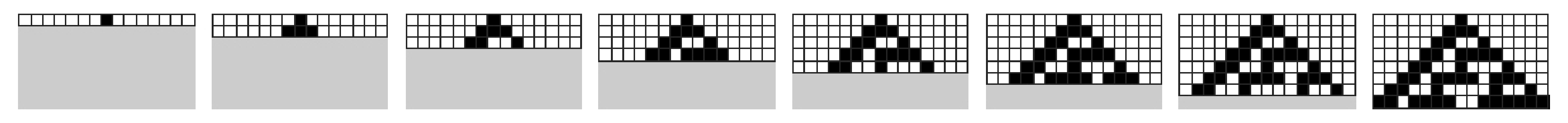
in this case a cell is made black if either of its neighbors – but not both – was black on the step before

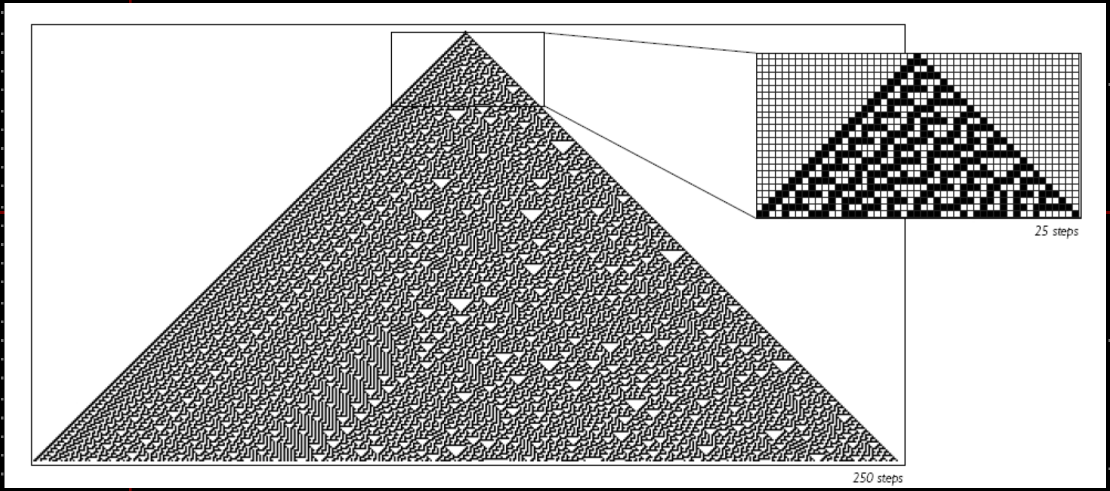
although intricate, this pattern actually consists of many nested triangular pieces

Some Wolfram Cellular Automaton Rules

rule 30







Examine a given cell and its immediate right neighbor. If both were white on the previous step, then the new color of the given cell should get the same color with the previous color of its immediate left neighbor. Otherwise it should get the opposite color of that

pattern of high complexity with almost no overall regularity

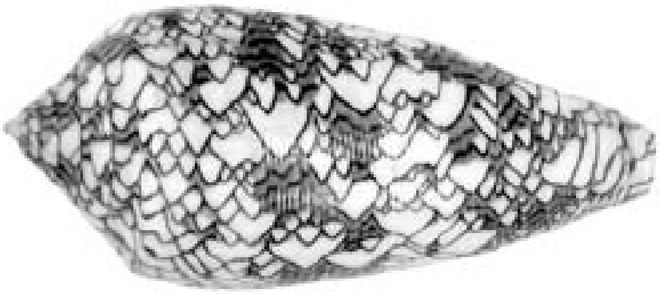
The new kind of science, which Wolfram develops in his book, analyses exactly this phenomenon: that even though the underlying rules of a system are simple, and even though the system is started from simple initial conditions, the behavior that the system shows can nevertheless be of high complexity. And he argues that this basic phenomenon is ultimately responsible for the most of the complexity observed in nature.

What is of great significance is that in most of the cases of the implementation of simple rules on cellular automata systems, it becomes almost impossible to predict the formation that will be produced. Indeed some of the patterns obtained show a remarkable mixture of regularity and irregularity. In many cases, as the system progresses, a variety of definite localized structures is produced. Some of these structures remain stationary, while others move steadily. On their own, each of these structures works in a fairly simple way, yet their various interactions can have an overall impact on the whole system. An example of such a behavior is rule 110. The behavior of the Game of Life is also similar to that of rule 110.

One is lead to question how it is possible for a simple rule to construct such complex systems, since the same rule affects each and every cell. But in fact the cells are not doing all the same thing. On the contrary, they seem to be doing quite different things. Some of them are part of the background, while others are part of a localized structure. According to Wolfram’s explanation, “what makes this possible is that even though individual cells follow the same rule, different configurations of cells with different sequences of colors can together produce all sorts of different kinds of behavior”.

This new kind of science analyses exactly this phenomenon: that even though the underlying rules of a system are simple, and even though the system is started from simple initial conditions, the behavior that the system shows can nevertheless be of high complexity. This basic phenomenon is ultimately responsible for the most of the complexity observed in nature.

Stephen Wolfram, A new Kind of Science, 2002



An example of Wolfram's results is that the complex pattern on this mollusc shell may just come from a simple program like a cellular automaton.

> “even though individual cells follow the same rule, different configurations of cells with different sequences of colors can together produce all sorts of different kinds of behavior ”.

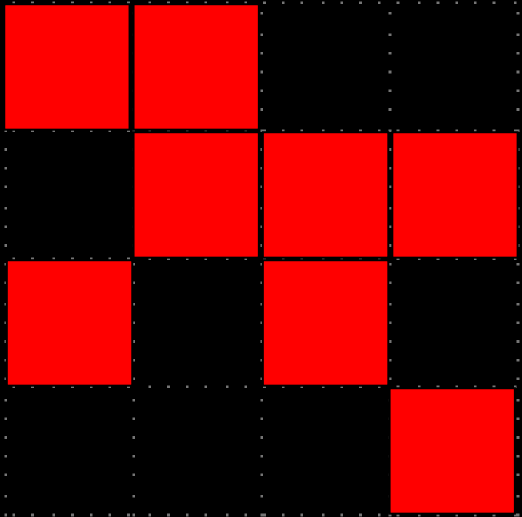
Stephen Wolfram, A new Kind of Science, 2002

Typical Features of Cellular Automata A4

1. absence of external control (autonomy)
2. symmetry breaking (loss of freedom / heterogeneity)
3. global order (emergence from local interactions)
4. self–maintenance (repair / reproduction metabolisms)
5. adaptation (functionality / tracking of external variations)
6. complexity (multiple concurrent values or objectives
7. hierarchy (multiple nested self-organized levels)

Applications of Cellular Automata

1. approximation of any physical system
2. lattice models for solidification and aggregation
3. structural mechanics
4. image processing
5. modeling of biological systems
6. modeling of simple behavior and functioning of organisms
7. study of chemical and physical turbulence
8. study of problems in number theory , tapestry design
9. forestry
10. urban planning, architecture and system dynamics



Artificial Life and Cellular Automata

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Alexandra Stamou

caad DARCH

> Artificial Life = “Life made by Man rather than by Nature ”
>
> Christopher G. Langton

Rather than studying biological phenomena by taking living organisms apart to see how they work, the main focus now is to recreate these phenomena from scratch within computers, by designing and examining systems, that behave like living organisms. In other words the field of Artificial Life revolves around the creation of life-like behavior generators.

More generally, the study of cellular automata is associated with the need to better understand artificial life. AL is a field of science that examines real life and the behavior of living species through computer models. Christopher G. Langton defines the notion Artificial Life as “Life made by Man rather than by Nature”. Rather than studying biological phenomena by taking living organisms apart to see how they work, the main focus now is to recreate these phenomena from scratch within computers, by designing and examining systems, that behave like living organisms. In other words the field of Artificial Life revolves around the creation of life-like behavior generators. It is though of great significance the realization that in terms of programming these generators, it is not possible to fully predict and specify the sequence of transitions that these machines will undergo. In general, as Langton underlines, “we cannot derive behaviors from specifications nor can we derive specifications from behaviors”. This means that in order to examine and determine the behavior of some machines, there is no resource but to run them and see how they behave. A13 The examination of cellular automata concludes to the fact that these mathematical models are good examples of the kind of computational paradigm Artificial Life seeks to achieve: bottom-up, parallel, local-determination of behavior. More specifically, cellular automata, as well as Lindenmayer Systems, flocking Boids and other similar systems, are considered to be recursively generated. The main characteristic of such structures is that they consist of sub-parts, modifiable by the rules of the system. Accordingly, these rules are usually sensitive to the context or the so-called “neighborhood” in which the sub-parts are embedded. Moreover there is no global rule applied to the entire structure, no global information is distributed in the system.

It is not possible to fully predict and specify the sequence of transitions that these machines will undergo.

in order to examine and determine the behavior of some machines, there is no resource but to run them and see how they behave.

"We cannot derive behaviors from specifications nor can we derive specifications from behaviors" .

Christopher G. Langton

The examination of cellular automata concludes to the fact that these mathematical models are good examples of the kind of computational paradigm Artificial Life seeks to achieve: bottom-up, parallel, local-determination of behavior

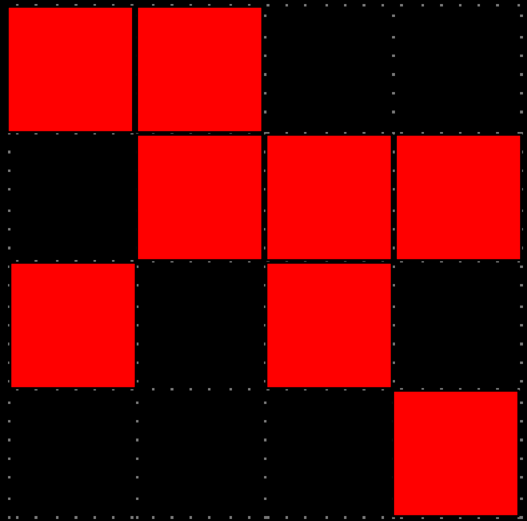
For example in a cellular automata’s structure, the transition function is applied to and obeyed homogeneously by each sub-part in the lattice, thus performing a local behavior in a simple discrete space/time universe. Subsequently this entire universe is updated, by applying this behavior to each cell over and over again. Throughout this recursive process all different kinds of context-sensitive rules can be embedded in the sub-parts of the universe, resulting to the computation and construction of new structures, which may again compute and construct. In the end, recursive, bottom-up specifications, used in the field of Artificial Life, lead to much more natural and flexible behavior at the global level, than the typical top-down specifications, used in the field of Artificial Intelligence.

In this context the feedback between the local and the global levels is of great importance. According to Langton, “the interactions among the low-level entities give rise to the global level dynamics which, in turn, affects the lower-levels by setting a local context within which each entity’s rules are invoked. Thus, local behavior supports global dynamics, which shapes local context, which affects local behavior, which supports global dynamics and so forth”. A13

the feedback between the local and
the global levels is of great importance

the interactions among the low-level entities give rise to the global level dynamics which, in turn, affects the lower-levels by setting a local context within which each entity's rules are invoked.

> “Local behavior supports global dynamics, which shapes local context, which affects local behavior, which supports global dynamics and so forth”.
>
> Christopher G. Langton



Architecture, Urban Design and Cellular Automata

Alexandra Stamou

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

caad DARCH

“More and more … the spaces of everyday life come loaded up with software, lines of code that are installing a new kind of automatically reproduced background and whose nature is only now starting to become clear”.

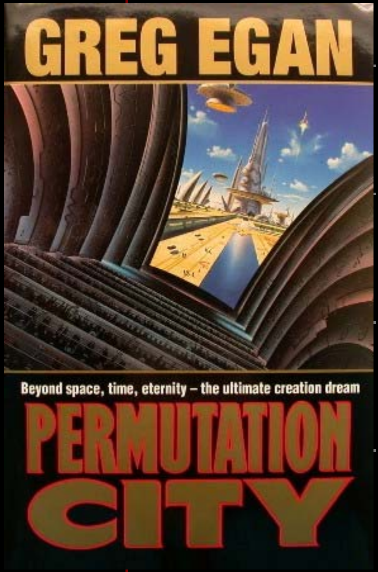
Thrift and French, 2002

The use of cellular automata does not have boarders. Their implementation can be met in various different fields. Greg Egan, an Australian cyberpunk writer, in his novel “Permutation City” addresses the question, if there is any difference between a perfect computer simulation and a “real” person. He argues that our universe could be but an algorithm running without the need of any physical substance. An interesting idea expressed in the novel is that of the “Autoverse”, which is an artificial life simulator ultimately based on a cellular automaton, complex enough to represent the substance of an artificial chemistry. The Autoverse is a chemistry set unclearly resembling real chemistry. In the novel, tiny environments simulated in the autoverse are filled with small populations of a simple artificial life form. These are opposed to a universe of huge virtual realities which operate under different heuristic rules, and therefore are incoherent. B31, B32

While cellular automata were originally developed to describe organic self-replicating systems, their structure and behavior were also useful in addressing architectural, landscape and urban design problems. The range of topics varies from vernacular settlements and social interaction to material behavior and air circulation.A4 For example there is a great variety of research papers and books that examine the exploration of the simulation of urban growth using complex systems theory and cellular automata.

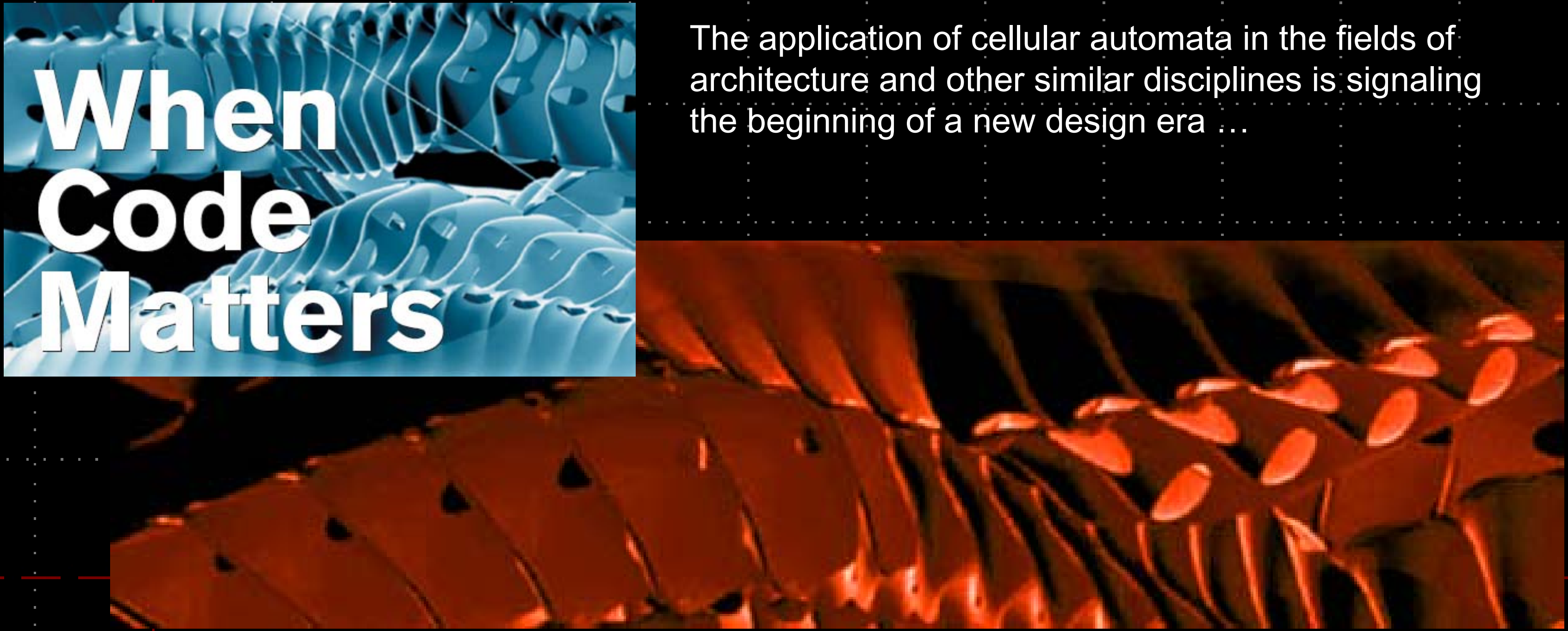
Until very recently the architectural design process always depended on the decision of what the architect would like to have as a final result. In order to achieve a specific final result the architect would have to specify from the beginning all design rules and methods, which should be followed and carried out until the last project was actually realized.

Greg Egan and "Permutation City"



In this novel, Egan describes the "Autoverse", which is an artificial life simulator ultimately based on a cellular automaton, complex enough to represent the substance of an artificial chemistry

Tiny environments simulated in the "Autoverse" are filled with small populations of a simple artificial life form. These are opposed to a universe of huge virtual realities which operate under different heuristic rules, and therefore are incoherent…



The application of cellular automata in the fields of architecture and other similar disciplines is signaling the beginning of a new design era …

The application of cellular automata in the fields of architecture and other similar disciplines is signaling the beginning of a new design era, where the architect sets a number of rules at the beginning, but is no longer in control of the final result, on the contrary a new kind of architecture emerges, where the final result cannot be predicted. While supporting the necessity of the implementation of cellular automata in the way architecture can be produced, one can refer to the fact that, although normally the design process starts from whatever the final result should be, yet to produce this result reliably, the designer has to restrict himself to systems and rules, whose “behavior” can already from the beginning understand and predict. Otherwise the final result cannot be foreseen, as the architect does not have control over the design process and the application of the design rules.

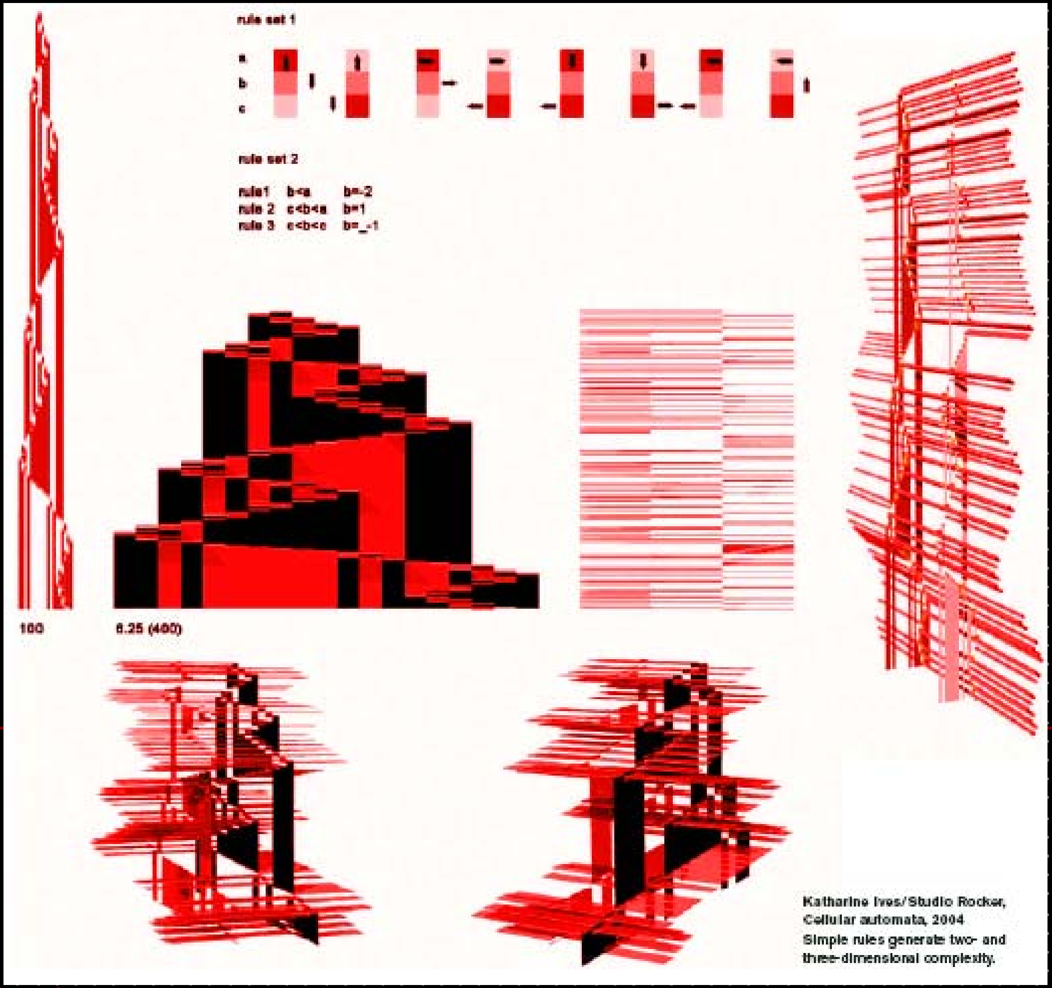
As Karl Chu points out in his article “Metaphysics of Genetic Architecture and Computation”, we now for the first time are able to think of a new kind of “xenoarchitecture” with its own autonomy and will to being. This new concept of architecture is adequate to the demands imposed by computation. To realize this concept, architecture has still yet to incorporate the architecture of computation into the computation of architecture. Contrary to Mies van der Rohe’s remark that architecture is the art of putting two bricks together, the emerging conception is that architecture is the art of putting two bits together, at least bits that are programmed to self-replicate, self-organize and self-synthesize into evermore new constellations of emergent relations and aggregations.

This way of producing structures in space brings us to the definition of an architecture, where each bit can be seen as a monad, a singular entity, at the most irreducible level, and by extension as a unit of a self-replicating system. It is often referred to as genetic architecture, and its theoretical origins can be traced to John von Neumann’s invention of the cellular automaton and his “von Neumann architecture” for self-replicating systems, which actually incorporates the ability of a system to contain a complete description of itself and use that information to create new copies.

In this case architecture is the construction of possible worlds generated by the universe of computational monads. Each monad encapsulates an internal principle that is generative, and each generative system transmits and propagates hereditary information. Each monad is at once a self-replicating and self-organizing system capable of constituting itself into a well integrated whole or a possible world. Karl Chu refers to that as “monadology of genetic architecture”.A9

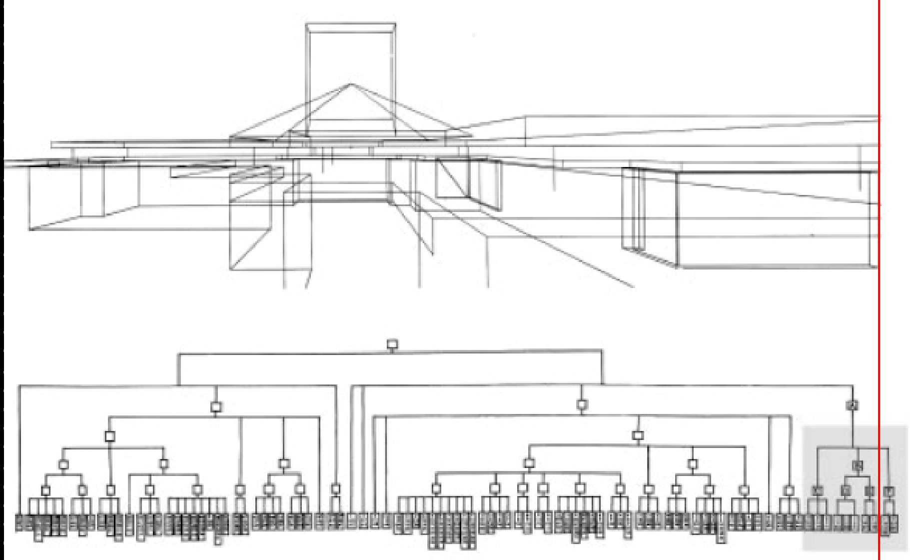
An example of an application of cellular automata in architecture is a competition entry of Mike Silver Architects for the San Jose State University Museum of Art and Design, in Silicon Valley, California, in 2003. This design team created a software, with the name Automason Ver 1.0, developed around the analogy between the operation of cellular automaton programs and masonry construction, in order to effect meaningful changes in the way work is produced in the field. The idea of using simple programs to drive the construction of bricks-and-mortar structures comes from the observation that masons work much like cellular automaton programs. By following procedures based on laws of adjacency and iteration, a mason builds by stacking one brick at a time. The use of cellular automaton programs can actually facilitate the production of extremely difficult designs without forcing the mason to do more work. What is more important is that in the case of applying simple programs to the design of possible structures, there is no need for a reduction to predetermined or ideal types. The only way to know how a given rule will behave is to set it in motion.

With the use of this software, all building details can obtain their complexity for free: no external agency is needed to design them. Such complexity is only dependent on the application of fixed rules in a discrete system. The overall form of a cellular automata masonry structure must therefore be evaluated in terms of its relationship to specific building requirements interpreted and organized by the architect. Moreover the functional constraints are used to willfully select self-organizing patterns that are particular to the computational properties of a specific material.A10 a new kind of architecture emerges, where the final result cannot be predicted

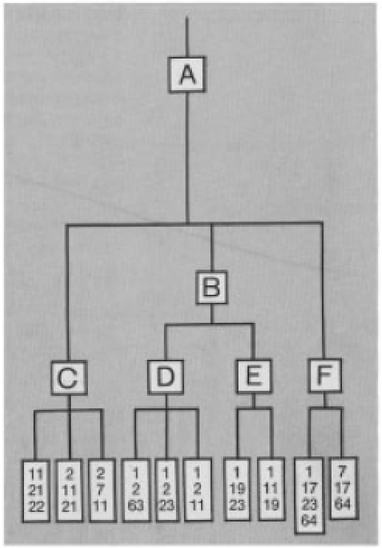


Katharine Ives/Studio Rocker,
Cellular automata, 2004
Simple rules generate two- and
three-dimensional complexity.

Katharine Ives, 2004
Studio Rocker, Cellular Automata

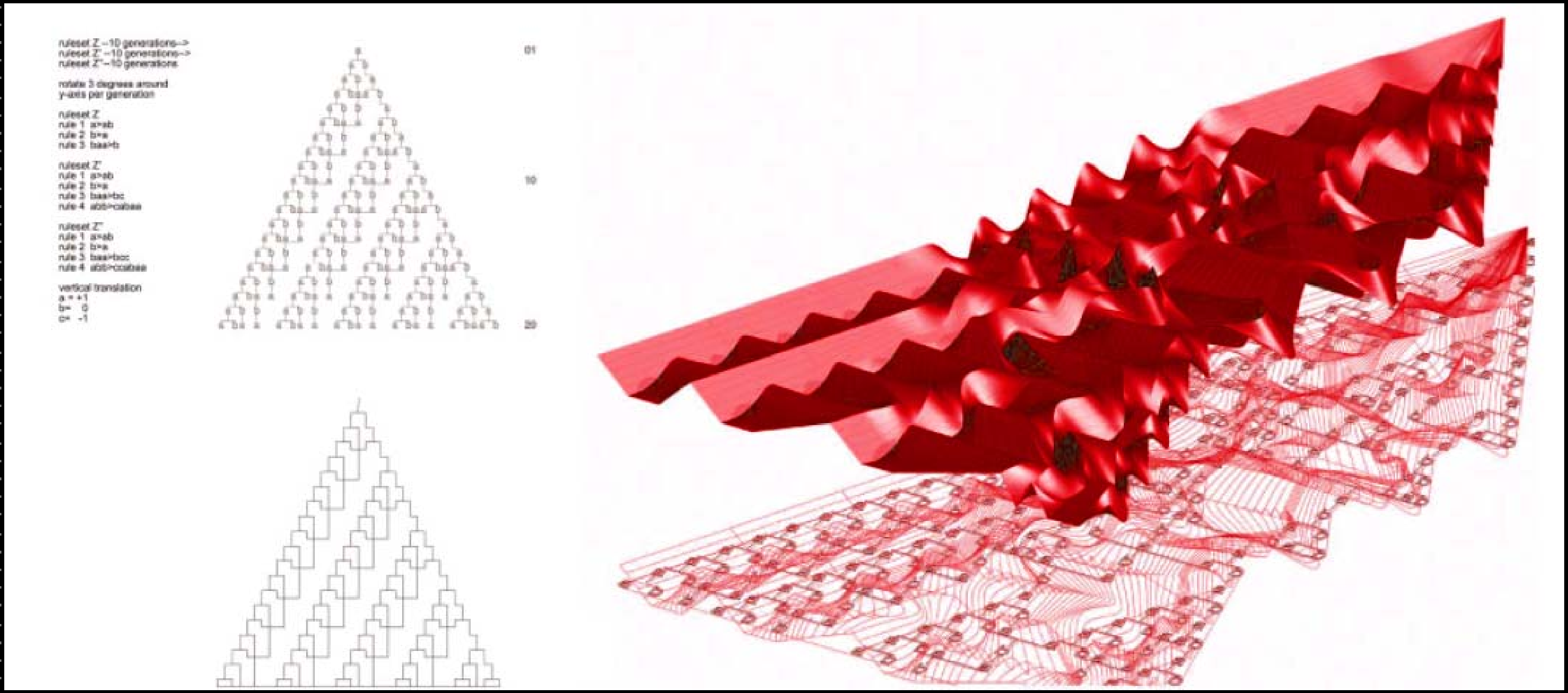


Algorithm House



Algorithm Garage

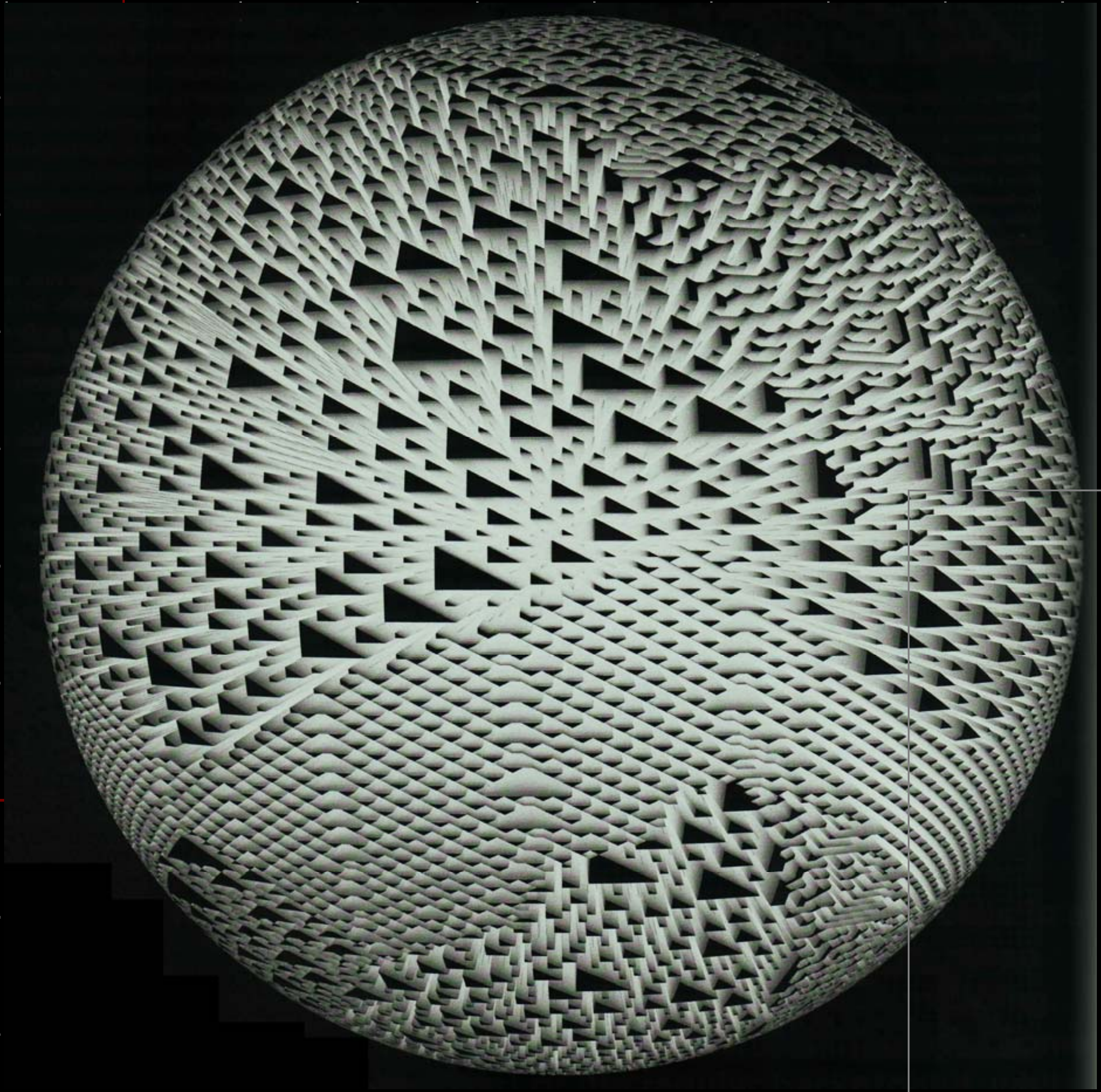
> “architecture has still yet to incorporate the architecture of computation into the computation of architecture…”
>
> Karl Chu



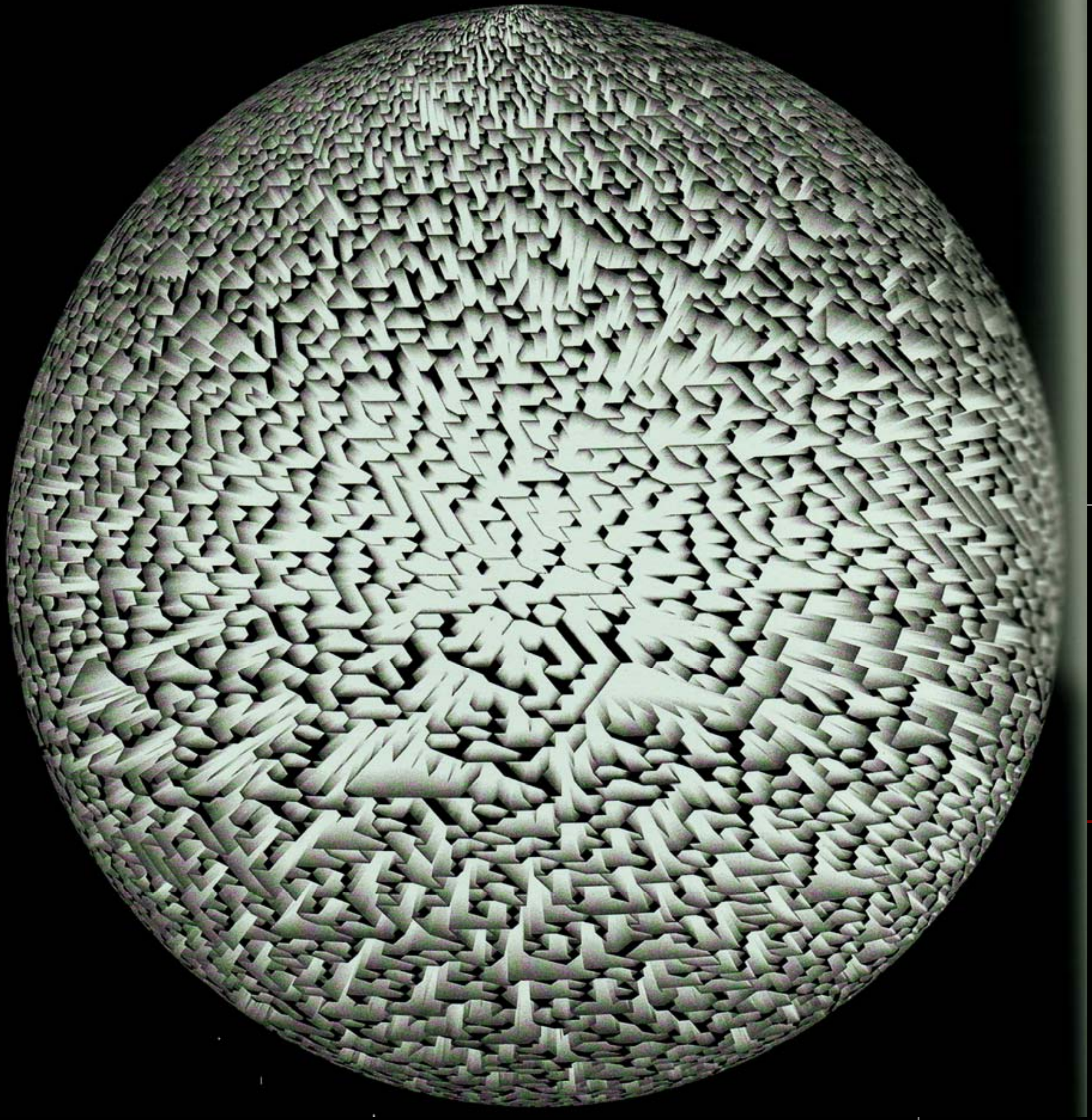
Brandon Williams, 2004
Studio Rocker, Recursions

"Contrary to Mies van der Rohe's remark that architecture is the art of putting two bricks together, the emerging conception is that architecture is the art of putting two bits together."

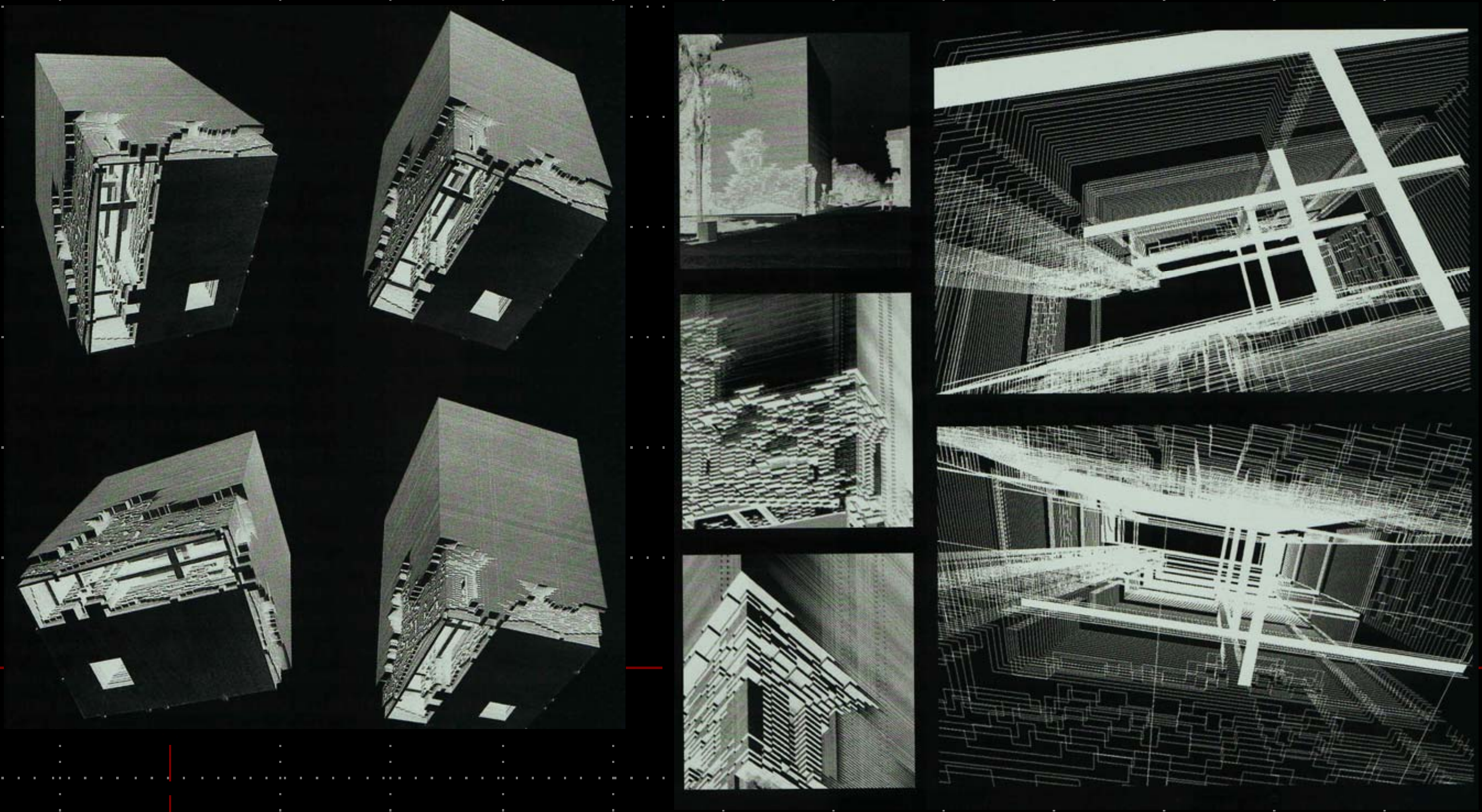
Karl Chu



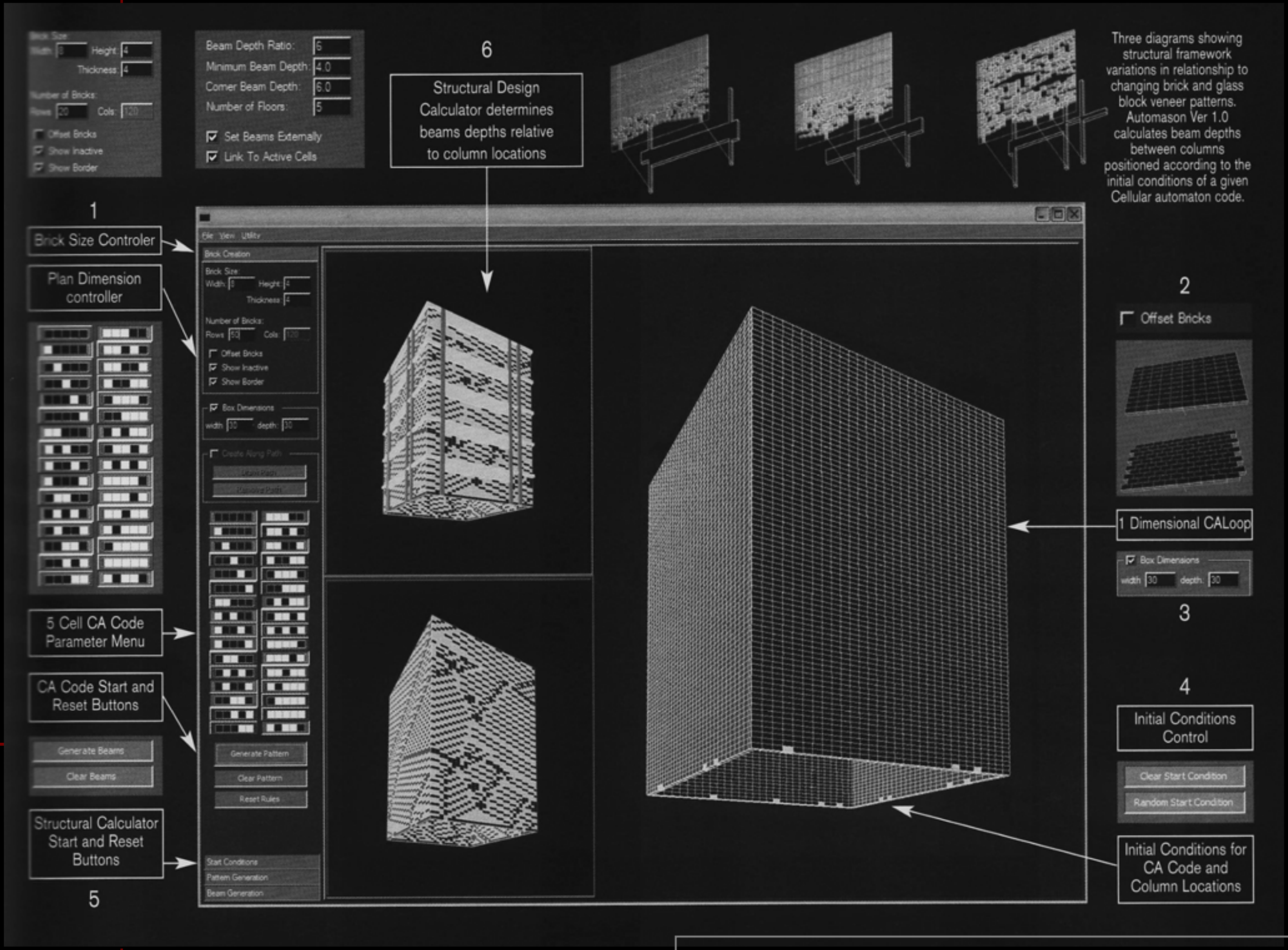
architecture is the construction of possible worlds generated by the universe of computational monads.



Each monad encapsulates an internal principle that is generative, and each generative system transmits and propagates hereditary information. Each monad is at once a self-replicating and self-organizing system capable of constituting itself into a well integrated whole or a possible world



Mike Silver Architects for the San Jose State University Museum of Art and Design, in Silicon Valley, California, in 2003

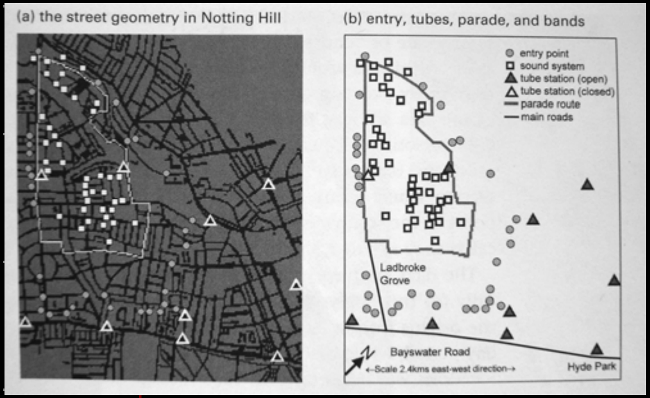


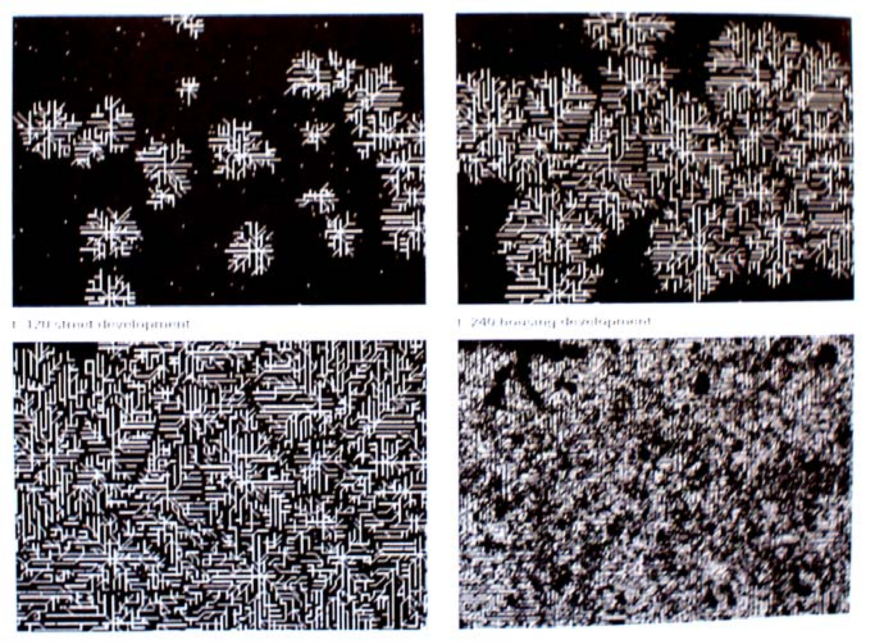
Three diagrams showing structural framework variations in relationship to changing brick and glass block veneer patterns. Automason Ver 1.0 calculates beam depths between columns positioned according to the initial conditions of a given Cellular automaton code.

Mike Silver Architects for the San Jose State University Museum of Art and Design, in Silicon Valley, California, in 2003

The design team of Mike Silver Architects used a specific cellular automata code to produce both complex and simple patterns from straight courses of stone-and-glass blocks. The functional and design requirements, such as rooms with windows or galleries requiring large, blank display walls, were laid out in accordance with the competition brief. As the team explains, “once these parameters were set in place, a search was made through multiple iterations in order to find the most appropriate patterns. For the museum’s exterior, internal subdivisions and fire stairs, a five-cell totalistic cellular automaton was found to create a partly windowless volume with intricate openings at the base. (…) Vertical supports for the building’s interior spaces were determined by the initial conditions of the CA code on the ground floor. A non-regular grid of columns produced different spans with beams of varying depths setting up an exchange between light, gravity and computation. (…) In the San Jose Museum, the nature and position of each masonry-unit affects its immediate neighbors and the order of the whole. Because the system is extremely sensitive to small changes, every brick counts in a truly organic architecture created by the application of simple programs”. This means that the pattern variations that are emerging are extremely sensitive to small errors. To avoid errors like that the design team also proposed the use of a typical mobile phone, which could be programmed to alert masons if any mistakes were made in the process of stacking blocks.A10

If we now move from the architectural level to the urban design level, then there is even a larger variety of research on the application of cellular automaton systems to the study of urban growth and evolution. But what is the way that the theory of CA can be applied to urban formations and transformations?





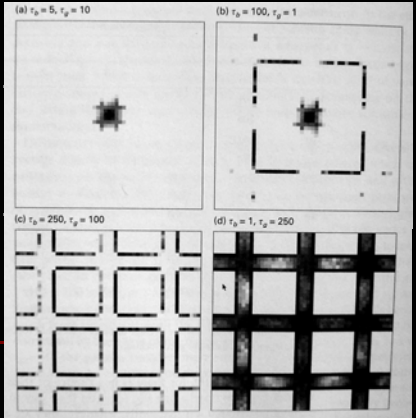


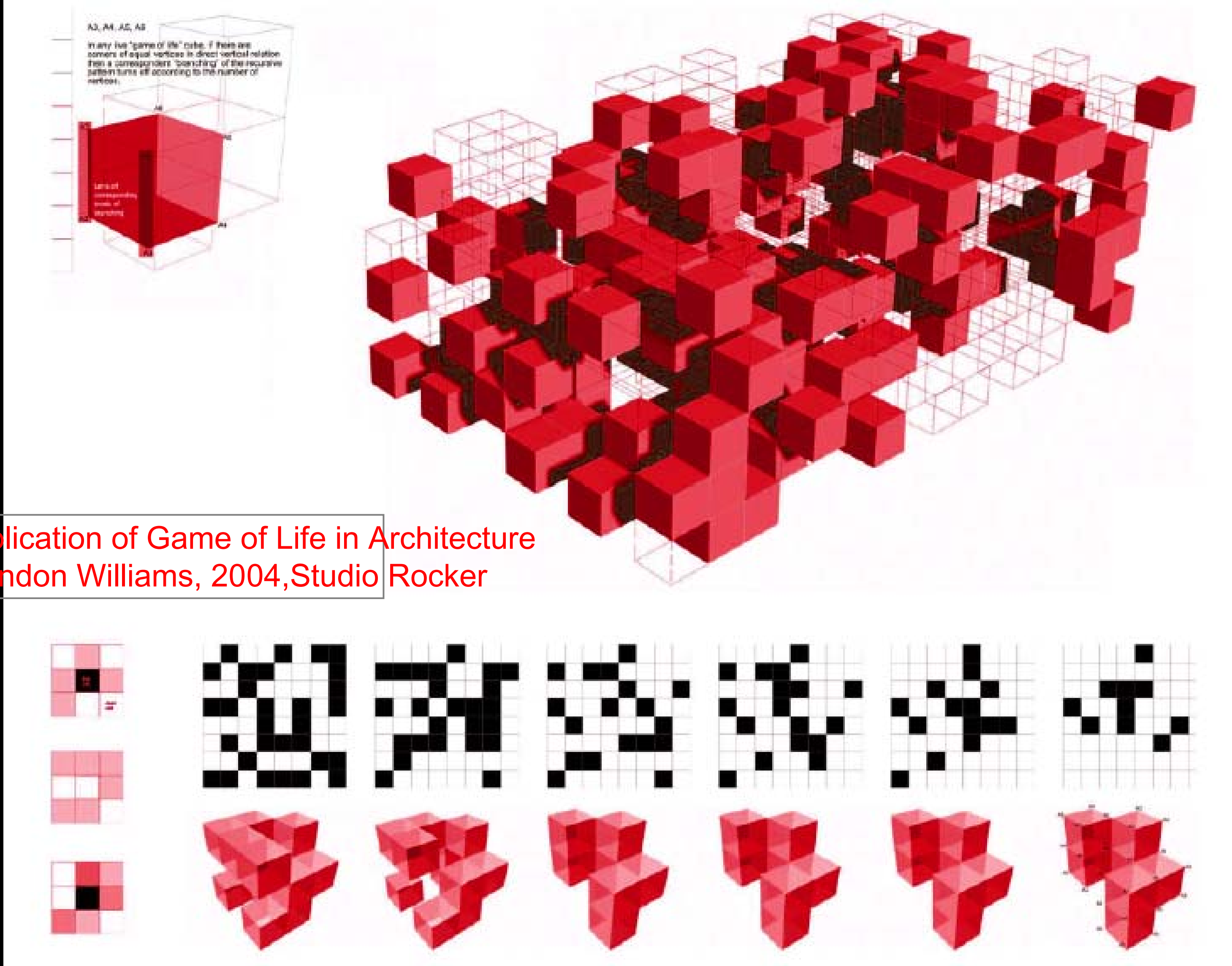
Figure 6.6
Variation in flow densities.

Urban development, urban models based on cellular automata and multi-agent systems

In 1961, Jane Jacobs, in her book “The Death and Life of Great American Cities” wrote that “Cities happen to be problems in organized complexity, like the life sciences. They present situations in which half a dozen or several dozen of quantities are all varying simultaneously and in subtly interconnected ways… The variables are many but are “interrelated into an organic hole”. Jacobs through many of her books played an important role to the foundation of a culture, where planning and design emerge from the bottom up, where individuals are in touch with the problems of the city and know best how to tackle them: a planning and design which is highly decentralized, in tune with the ways cities grow – from the bottom up.A19 Michael Batty in his book “Cities and Complexity” refers that Jacob drew her inspiration from Warren Weaver’s (1948) address to the Rockefeller Foundation, in which he suggested that systems could be classified as applicable to three kinds of problem: problems of simplicity, problems of disorganized complexity and problems of organized complexity. It is the last category that Weaver argued should form the cutting edge of science. Jane Jacobs was the first to propose that these “problems of organized complexity” should include cities. This notion was later also supported by others, such like Christopher Alexander, who argued that only if we see cities in this bottom-up way, will we ever be able to develop a comprehensive understanding of them.A19

In his book Michael Batty implements cellular automata theory into simple city models, in order to simulate such a bottom-up emergence. And he does so by representing the basic elements of the city in two distinct but related ways: through cells, which represent the physical and spatial structure of the city, and through agents, which represent the human and social units that make the city work. The category of the cells is studied with the help of cellular automata, whereas the category of the agents is studied through the scope of multi-agent systems. Cells are fixed, Agents are mobile and move between locations. By studying an urban system in this way, “concepts of criticality, threshold, surprise, novelty and phase transition are introduced in the context of spatial development. These ideas are finally synthesized in urban morphology, which reflects notions of local action and global pattern through self-similarity and fractal geometry”. In an urban environment of a city the most elementary forms of dynamics can generate complexity in the form of chaos. “This highlights the fact that we are dealing with systems that are far-from-equilibrium, where the structures that we usually observe are highly ordered but in the “edge-of-chaos”. To illustrate this idea Batty is describing a series of spatial models, which he implements as cellular automata. Self-organization, self-similarity and segregation all feature in these models.

Emergence in urban systems can now be viewed as the successive operation of processes defined by cellular automata. It can be specified as a combination of three effects that determine change – positive feedback, innovation and interaction. All these effects operate locally: the first two are entirely localized to the individual or cell in question and the third operates on adjacent locations. Although nothing here suggests the possibility of the emergence of a global pattern, nevertheless eventually, if the system is growing, these locations will interact through the spreading CA effect.A19



Application of Game of Life in Architecture
Brandon Williams, 2004,Studio Rocker

Turning the discussion back again to the field of architecture, Ayşe Erzan remarks that architectural design is one realization of a “machine” that once set in motion, produces different complex behaviors identified by the functions that occur in this space of human activity. Erzan argues that the fact that both periodic and complex spatio-temporal patterns can be generated by the same cellular automaton gives rise to the possibility that the complexity of architectural design can likewise be parameterized and the function to be generated coded in a universal machine.

Cellular automata afford economically storable and classifiable sets of rules, like diagrams or abstract machines, which acting on an initial configuration, they are able to generate periodic or endlessly evolving sequences of shapes. The abstract machine can be used to model both finished buildings and the functions that will be discharged in and by them. Whether or not these sequences are pleasant, useful etc., is a matter of the architect experimenting with different parameters and configurations. But they suggest new ways to enlarge our architectural vocabulary. A1

A. Related Literature and Reference

1. Erzan, Ayşe: “*Abstract Machines and Calculable Grammar of geometrical Shapes*”, in ”Anytime”, edited by Davidson, Cynthia C., The MIT Press, Cambridge 1999
2. Richter, Klaus, Rost, Jan-Michael: ”*Komplexe Phänomene durch Interaktion: Zelluläre Automaten*”, in “*Komplexe Systeme*”, Fischer Kompakt, Frankfurt am Main 2004
3. Richter, Klaus, Rost, Jan-Michael: “*Zelluläre Automaten*”, in “*Komplexe Systeme*”, Fischer Kompakt, Frankfurt am Main 2004
4. Terzidis, Kostas: “*Algorithmic Architecture*”, Architectural Press, Elsevier, Cambridge 2006
5. Rocker, Ingeborg M.: “*When Code Matters*”, in Architectural Design, Vol.76, Issue 4, “*Programming Cultures*”, Jul/Aug 2006
6. Wolfram, Stephen: “*How Do Simple Programs Behave*”, in Architectural Design, Vol.76, Issue 4, “*Programming Cultures*”, Jul/Aug 2006
7. Wolfram, Stephen: “*Cellular Automata and Complexity, Collected Papers*”, Addison – Wesley Publishing Company, USA 1994
8. Chopard, Bastien, Droz, Michel: “*Cellular Automata Modeling of Physical Systems*”, Cambridge University Press, NY 2005
9. Chu, Karl: “*Metaphysics of Genetic Architecture and Computation*”, in Architectural Design, Vol.76, Issue 4, “*Programming Cultures*”, Jul/Aug 2006
10. Silver, Mike: “Building Without Drawings: Automason Ver 1.0”, in Architectural Design, Vol.76, Issue 4, “*Programming Cultures*”, Jul/Aug 2006
11. Wolfram, Stephen: “*The World of Simple Programs*”, in “A New Kind Of Science”, Wolfram Media Incorporation, 2002
12. Hight, Christopher, Perry, Chris: “*Collective Intelligence in Design*”, Introduction to Architectural Design, Vol.76, Issue 5,”*Collective Intelligence in Design*”. Sept/Oct 2006
13. Langton, Christopher G.: “*Artificial Life*”, Addison – Wesley Publishing Company, USA 1989

A. Related Literature and Reference

14. Egan, Greg: *“Permutation City”*, Australia, 1994
15. Negroponte, Nicolas: *“Towards a Humanism through Machines”*, in Architectural Design, Sept. 1969
16. Silver, Mike: *“Towards a programming Culture in the Design Arts”*, in Architectural Design, Vol.76, Issue 4, *“Programming Cultures”*, Jul/Aug 2006
17. McCullough, Malcolm: *“20 Years of Scripted Space”*, in Architectural Design, Vol.76, Issue 4, *“Programming Cultures”*, Jul/Aug 2006
18. Gardner, Martin: *“Mathematical Games: The Phantastic Combinations of John Conway’s new Solitaire Game “Life””*, in Scientific American 223, Oct 1970
19. Batty, Michael: *“Cities and Complexity: Understanding Cities with Cellular Automata, Agent-Based Models, and Fractals”*, The MIT Press, Cambridge, Massachusetts, 2005

B. Internet Resources

http://www.llnl.gov/computing/tutorials/parallel_comp/
http://en.wikipedia.org/wiki/Parallel_computing
http://www.answers.com/topic/cellular-automaton
http://www.answers.com/topic/conway-s-game-of-life
http://www.bitstorm.org/gameoflife/
http://www.radicaleye.com/lifepage/lifepage.html#otherpages
http://en.wikipedia.org/wiki/Conway's_Game_of_Life
http://processing.org/learning/examples/cellularautomata1.html
http://processing.org/learning/examples/wolframca.html
http://www.ibiblio.org/e-notes/Life/Game.htm
http://beat.doebe.li/bibliothek/w01401.html
http://beat.doebe.li/bibliothek/w01402.html
http://mathworld.wolfram.com/ElementaryCellularAutomaton.html
http://www.alvyray.com/Papers/PapersCA.htm
http://everything2.com/index.pl?node_id=1230445&lastnode_id=756640
http://www.soas2006.org/
http://books.google.com/books?hl=en&lr=&id=XK93Qx3fJCcC&oi=fnd&pg=PA1&sig=1zmQpeRikLdXdrJtVbexEvyS3UQ&dq=cellular+automat#PPP1,M1
http://en.wikipedia.org/wiki/Cellular_automaton
http://en.wikipedia.org/wiki/Emergence
http://ddi.cs.uni-potsdam.de/HyFISCH/Produzieren/lis_projekt/proj_gamelife/ConwayScientificAmerican.htm
http://timtyler.org/

B. Internet Resources

http://www.pixelsex.org/impressions.html
http://www.kunstfassade.de/index.html
http://www.biomodeling.info/simulator/simulator.html
http://www.wolframscience.com/downloads/basicimages.html
http://www.fraktalwelt.de/lsys/index.html
http://www.collidoscope.com/modernca/
http://www.mirwoj.opus.chelm.pl/ca/index.html
http://cell-auto.com/ising/index.html
http://www.complexcity.info/
http://www.cscs.umich.edu/~crshalizi/reviews/permutation-city/
http://gregegan.customer.netspace.net.au/
http://tecfa.unige.ch/perso/staf/nova/blog/index.php?s=cellular+automata
http://nwanua.aniomagic.com/media/domes_july_04.mov
http://www.decept.org/nolife/index_english.html
http://www.donhopkins.com/lzxnet/my-apps/vonNeumann/vonNeumann.lzx
http://stuff.mit.edu/afs/athena/course/4/4.272/www/CellularAutomata/sld002.htm
http://madeira.cc.hokudai.ac.jp/RD/takai/automa.html
http://www.santafe.edu/

C. Picture Resources

www.atomicarchive.com/Bios/vonNeumannPhoto.shtml
libai.math.ncu.edu.tw/bcc16/pool/3.02.shtml
www.abqarts.org/cultural/survey/polish-cs.htm
http://ddi.cs.uni-potsdam.de/HyFISCH/Produzieren/lis_projekt/proj_gamelife/ConwayScientificAmerican.htm
http://www.santafe.edu/research/topics-information-processing-computation.php
http://pro.corbis.com

- Rocker, Ingeborg M.: “*When Code Matters*”, in Architectural Design, Vol.76, Issue 4, “*Programming Cultures*”, Jul/Aug 2006
- Wolfram, Stephen: “*How Do Simple Programs Behave*”, in Architectural Design, Vol.76, Issue 4, “*Programming Cultures*”, Jul/Aug 2006
- Chu, Karl: “*Metaphysics of Genetic Architecture and Computation*”, in Architectural Design, Vol.76, Issue 4, “*Programming Cultures*”, Jul/Aug 2006
- Silver, Mike: “Building Without Drawings: Automason Ver 1.0”, in Architectural Design, Vol.76, Issue 4, “*Programming Cultures*”, Jul/Aug 2006
- Batty, Michael: “*Cities and Complexity: Understanding Cities with Cellular Automata, Agent-Based Models, and Fractals*”, The MIT Press, Cambridge, Massachusetts, 2005