Cellular Automata and the Quest for Artificial Self-Reproducing Structures

Markus Holzer Martin Kutrib

Institut für Informatik, Universität Giessen

Overview

- Fundamental Questions
- The Birth of Cellular Automata
- Non-Trivial Self-Reproduction
- Game of Life
- Synchronization
- Signals

Overview

- ➜ Fundamental Questions
- ➜ The Birth of Cellular Automata
- ➜ Non-Trivial Self-Reproduction
- ➜ Game of Life
- ➜ Synchronization
- ➜ Signals

Overview

- Fundamental Questions
- The Birth of Cellular Automata
- Non-Trivial Self-Reproduction
- Game of Life
- Synchronization
- Signals

Overview

- Fundamental Questions
- The Birth of Cellular Automata
- Non-Trivial Self-Reproduction
- Game of Life
- Synchronization
- Signals

Overview

- Fundamental Questions
- The Birth of Cellular Automata
- Non-Trivial Self-Reproduction
- Game of Life
- Synchronization
- Signals

Overview

- Fundamental Questions
- The Birth of Cellular Automata
- Non-Trivial Self-Reproduction
- Game of Life
- Synchronization
- Signals

Fundamental Questions

Fundamental questions [John von Neumann 1949]:

Fundamental Questions

Fundamental questions [John von Neumann 1949]:

(A) Logical universality. When is a class of automata logically universal? Also, with what additional attachments is a single automaton logically universal?

Fundamental Questions

Fundamental questions [John von Neumann 1949]:

(A) Logical universality. When is a class of automata logically universal? Also, with what additional attachments is a single automaton logically universal?

(B) Constructibility. Can an automaton be constructed by another automaton? What class of automata can be constructed by one, suitably given, automaton?

Fundamental Questions

Fundamental questions [John von Neumann 1949]:

(A) Logical universality. When is a class of automata logically universal? Also, with what additional attachments is a single automaton logically universal?

(B) Constructibility. Can an automaton be constructed by another automaton? What class of automata can be constructed by one, suitably given, automaton?

(C) Construction-universality. Can any one, suitably given, automaton be construction-universal, that is, be able to construct every other automaton?

Fundamental Questions

Fundamental questions [John von Neumann 1949]:

(D) Self-reproduction. Can any automaton construct other automata that are exactly like it? Can it be made, in addition, to perform further tasks?

Fundamental Questions

Fundamental questions [John von Neumann 1949]:

(D) Self-reproduction. Can any automaton construct other automata that are exactly like it? Can it be made, in addition, to perform further tasks?

(E) Evolution. Can the construction of automata by automata progress from simpler types to increasingly complicated types? Also, can this evolution go from less efficient to more efficient automata?

A self-reproducing machine is an artificial construct that is theoretically capable of autonomously manufacturing a copy of itself using raw materials taken from its environment.

A self-reproducing machine is an artificial construct that is theoretically capable of autonomously manufacturing a copy of itself using raw materials taken from its environment.

- What is an artificial construct?
- What is raw material?
- What is an environment?

Von Neumann's Kinematic Model:

- Von Neumann proposed a kinematic self-reproducing automaton model as a thought experiment [1948/1949].

Von Neumann's Kinematic Model:

- Von Neumann proposed a kinematic self-reproducing automaton model as a thought experiment [1948/1949].
- A hypothetical physical machine uses a storage of spare parts as its source of raw materials.

Von Neumann's Kinematic Model:

- Von Neumann proposed a kinematic self-reproducing automaton model as a thought experiment [1948/1949].
- A hypothetical physical machine uses a storage of spare parts as its source of raw materials.
- The machine has instructions written on a memory tape.

Von Neumann's Kinematic Model:

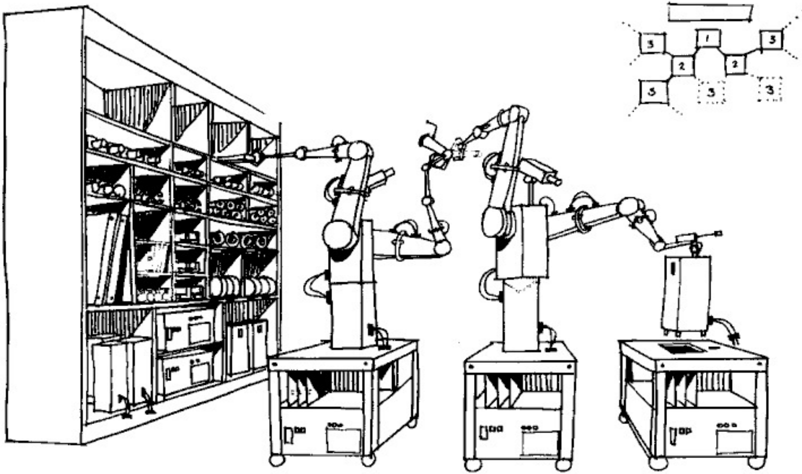
- Von Neumann proposed a kinematic self-reproducing automaton model as a thought experiment [1948/1949].
- A hypothetical physical machine uses a storage of spare parts as its source of raw materials.
- The machine has instructions written on a memory tape.
- The instructions direct it to take parts from the storage using a manipulator,

Von Neumann's Kinematic Model:

- Von Neumann proposed a kinematic self-reproducing automaton model as a thought experiment [1948/1949].
- A hypothetical physical machine uses a storage of spare parts as its source of raw materials.
- The machine has instructions written on a memory tape.
- The instructions direct it to take parts from the storage using a manipulator,
- assemble them into a duplicate of itself,

Von Neumann's Kinematic Model:

- Von Neumann proposed a kinematic self-reproducing automaton model as a thought experiment [1948/1949].
- A hypothetical physical machine uses a storage of spare parts as its source of raw materials.
- The machine has instructions written on a memory tape.
- The instructions direct it to take parts from the storage using a manipulator,
- assemble them into a duplicate of itself,
- and then copy the contents of its memory tape into the empty tape of the offspring.


3
1
3
2
2
3
3
3

The Birth of Cellular Automata

Artificial self-reproduction: Abstract from the natural self-reproduction problem its logical form.

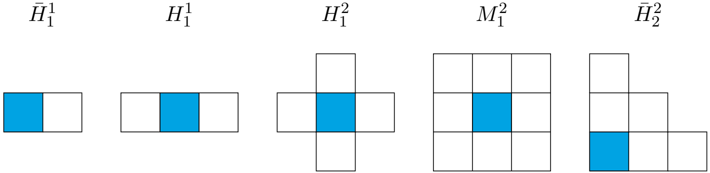
Von Neumann's Cellular Automata:

- Stanisław Ulam suggested to employ a mathematical device which is a multitude of interconnected machines operating in parallel to form a larger machine.

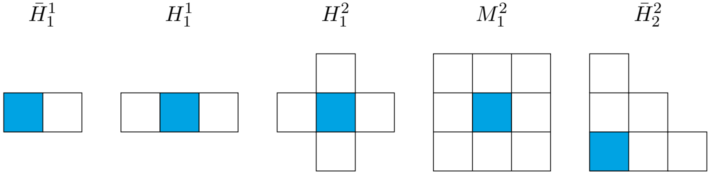
Von Neumann's Cellular Automata:

- Stanisław Ulam suggested to employ a mathematical device which is a multitude of interconnected machines operating in parallel to form a larger machine.
- Two-dimensional grid of cells (machines).
- Synchronous behavior.
- Cells are deterministic finite automata (simplicity).
- All cells are identical (homogeneity).
- One interconnection scheme (homogeneous local communication structures).

Interconnection schemes:



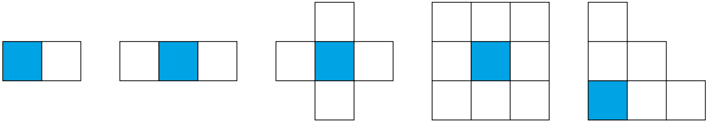
Interconnection schemes:



Quiescent state: If a cell itself and all of its neighbors are in the quiescent state, the cell remains in the quiescent state.

Interconnection schemes:

$\bar{H}_1^1$ H_1^1 H_1^2 M_1^2 $\bar{H}_2^2$



Quiescent state: If a cell itself and all of its neighbors are in the quiescent state, the cell remains in the quiescent state.

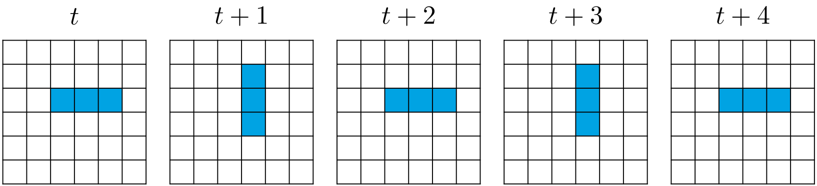
Larger machines are patterns of cell states, embedded in space.

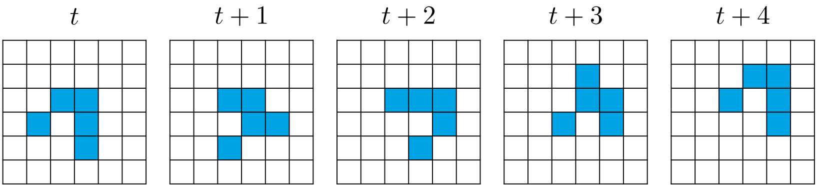
Example:

- The state set is $\{0, 1\}$.
- Each cell is connected to its eight immediate neighbors.

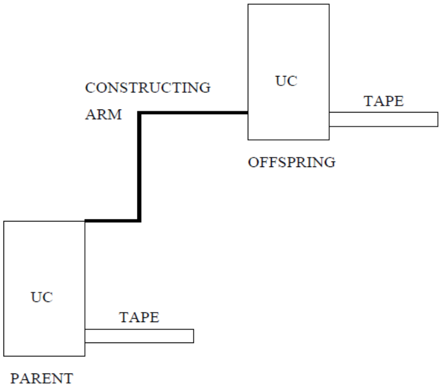
Example:

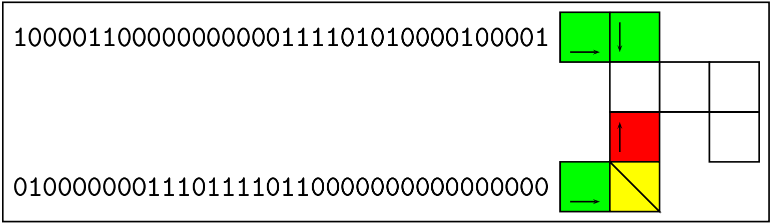
- The state set is $\{0, 1\}$.
- Each cell is connected to its eight immediate neighbors.
- The local transition function is defined by the sum of the states of the neighbors and of the cell itself. In particular:
- A cell enters state 1, if the sum is three.
- A cell keeps its current state, if the sum is four.
- Otherwise the cell enters state 0.


t
$t+1$
$t+2$
$t+3$
$t+4$

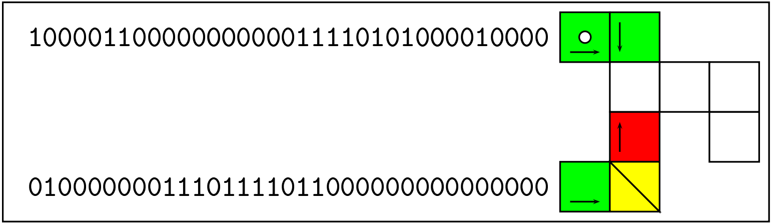

t
$t+1$
$t+2$
$t+3$
$t+4$

John von Neumann succeeded. He constructed a 29-state cellular automaton which is contruction-universal, self-reproducing, and Turing-universal.

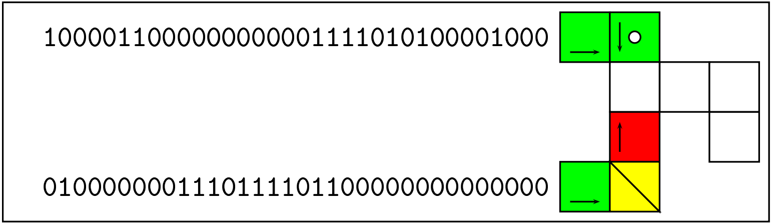


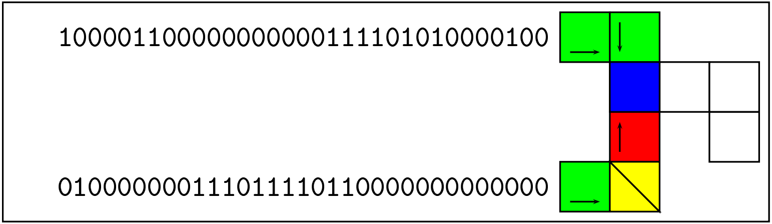


100001100000000000111101010000100001
010000000111011110110000000000000000

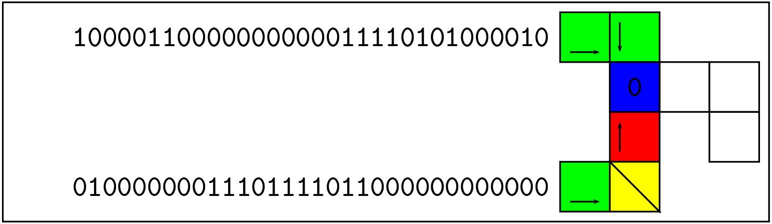


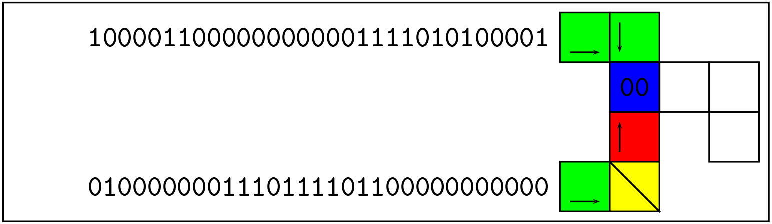
10000110000000000011110101000010000
01000000011101111011000000000000000

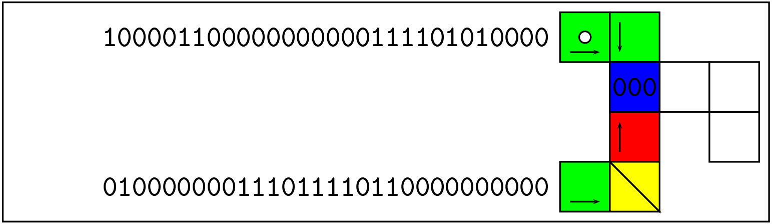

1000011000000000001111010100001000
0100000001110111101100000000000000



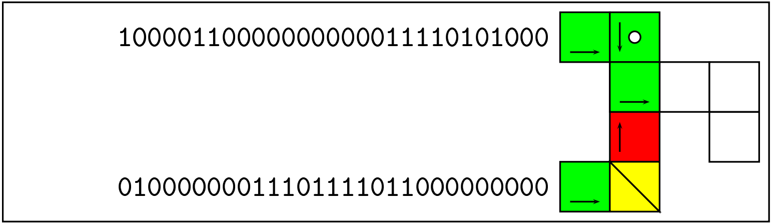
100001100000000000111101010000100
010000000111011110110000000000000

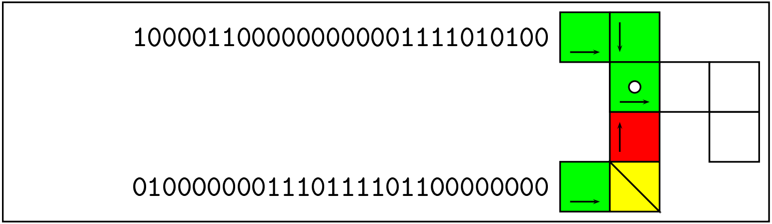

10000110000000000011110101000010
01000000011101111011000000000000
0


1000011000000000001111010100001
00
0100000001110111101100000000000

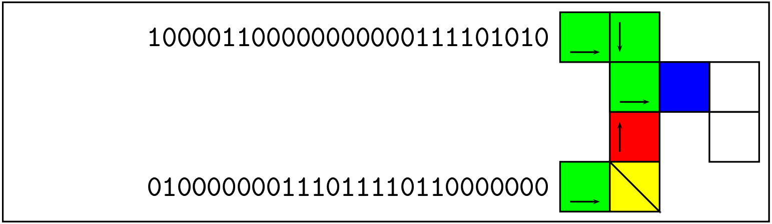


100001100000000000111101010000
000
010000000111011110110000000000

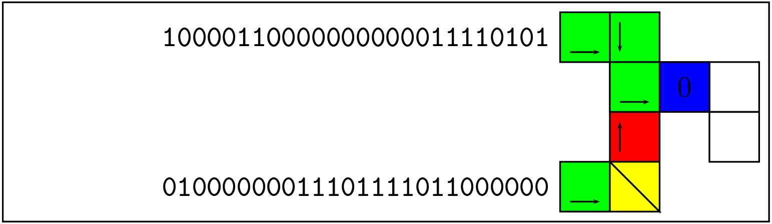

10000110000000000011110101000
01000000011101111011000000000

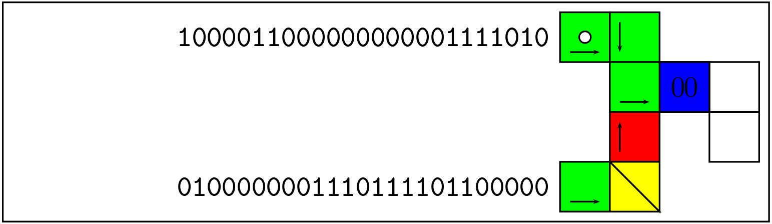


1000011000000000001111010100
0100000001110111101100000000

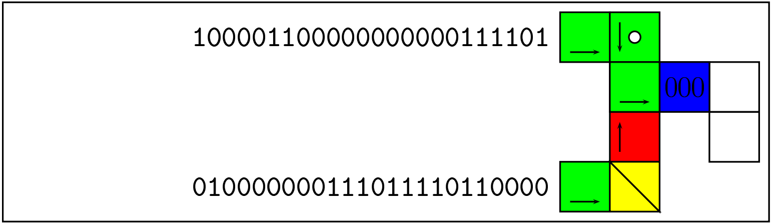


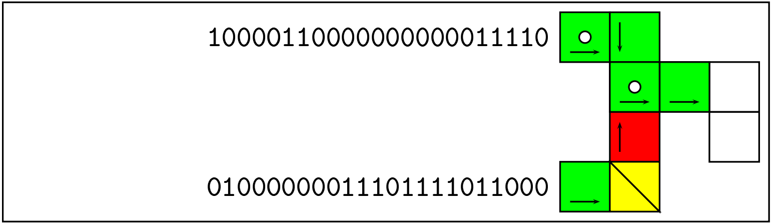
100001100000000000111101010
010000000111011110110000000

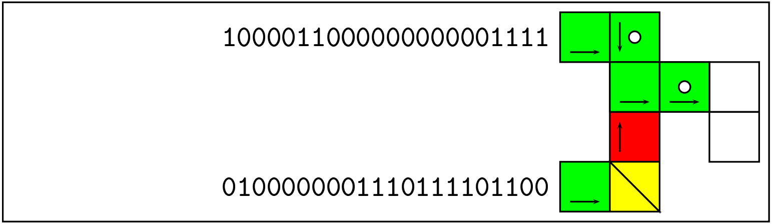

10000110000000000011110101
0
01000000011101111011000000

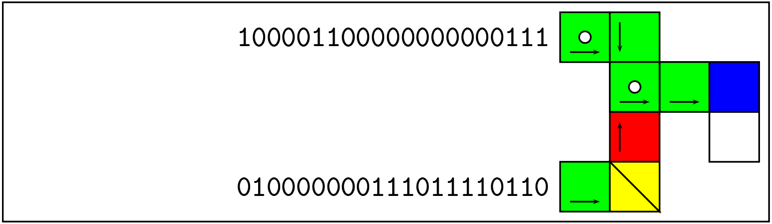


1000011000000000001111010
00
0100000001110111101100000

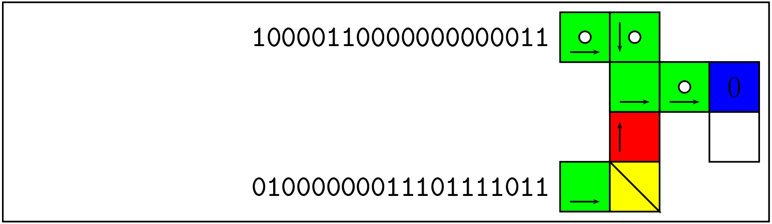

100001100000000000111101
000
010000000111011110110000

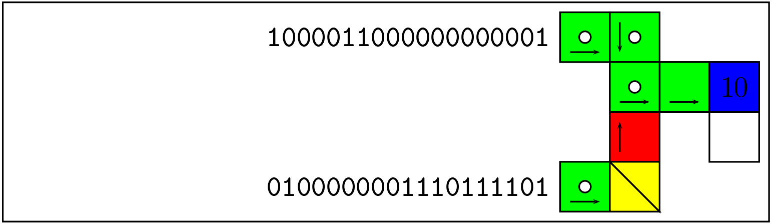

10000110000000000011110
01000000011101111011000

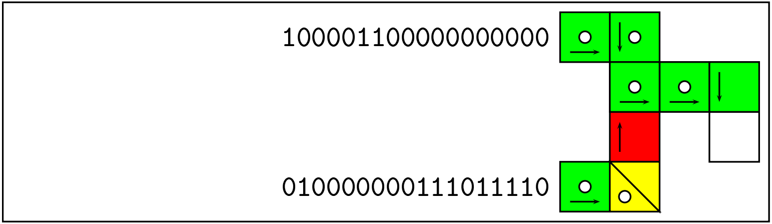

1000011000000000001111
0100000001110111101100

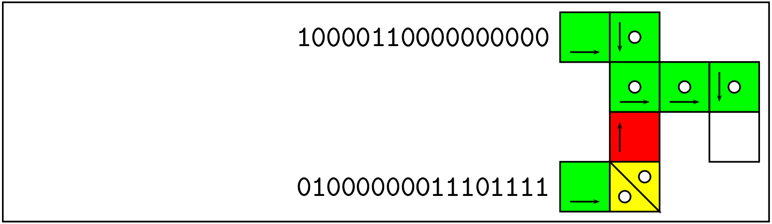


100001100000000000111
010000000111011110110

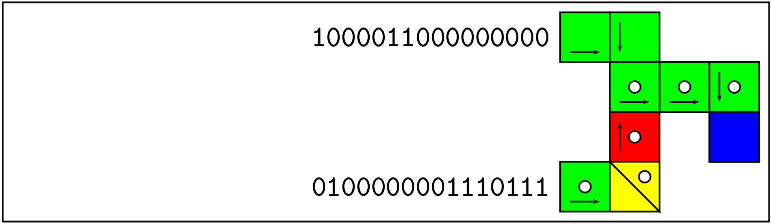

10000110000000000011
0
01000000011101111011

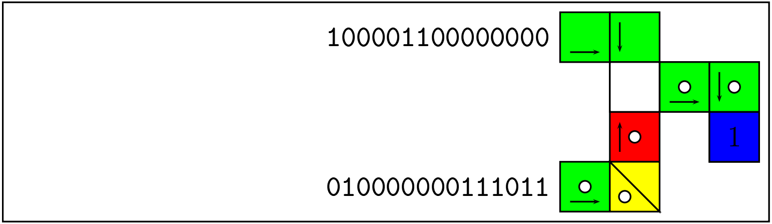

1000011000000000001
10
0100000001110111101

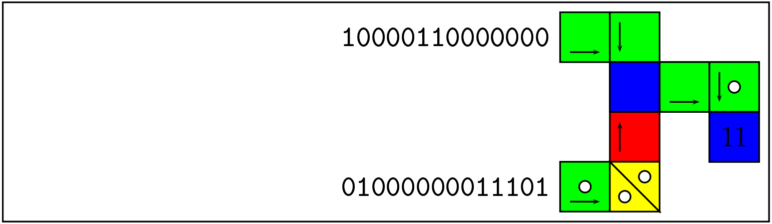

100001100000000000
010000000111011110

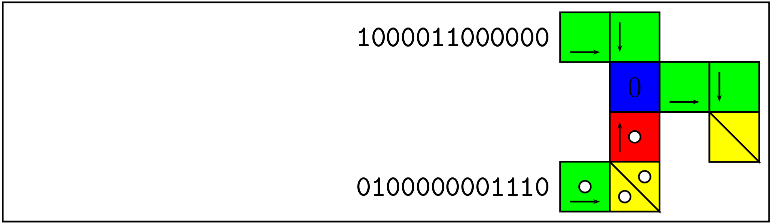


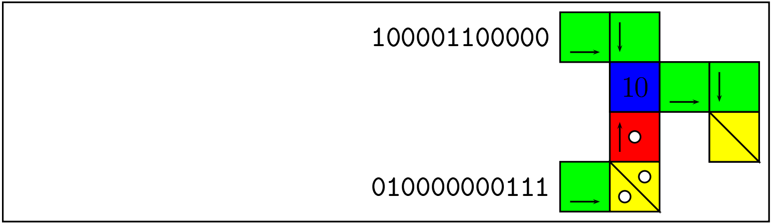
10000110000000000
01000000011101111

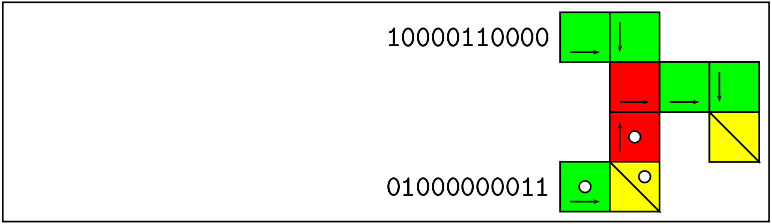

1000011000000000
0100000001110111


100001100000000
010000000111011
1

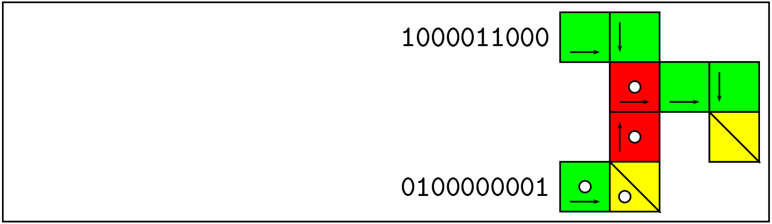

10000110000000
01000000011101
11

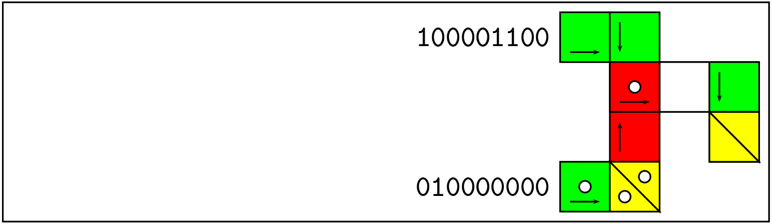

1000011000000
0
0100000001110


100001100000
10
010000000111

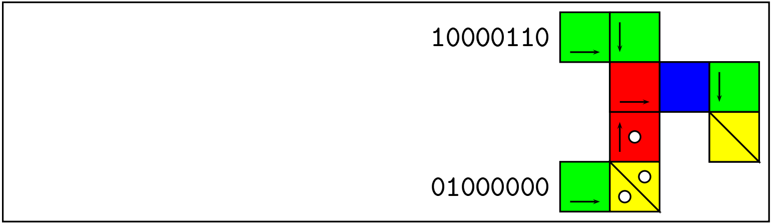


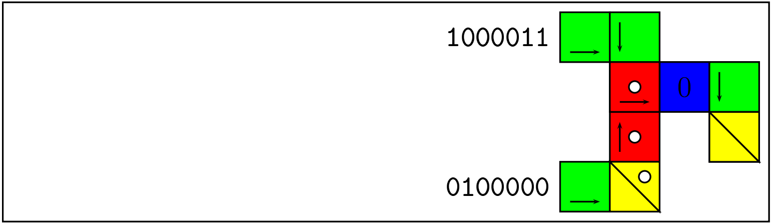
10000110000
01000000011

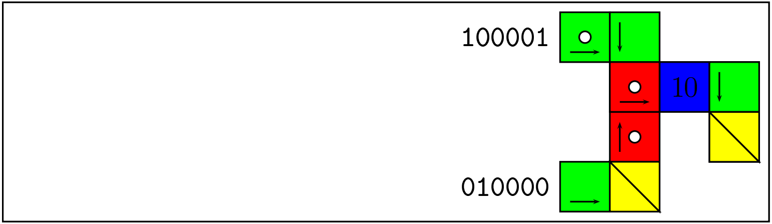

1000011000
0100000001

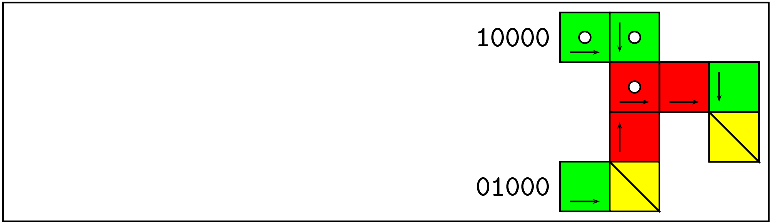


100001100
010000000

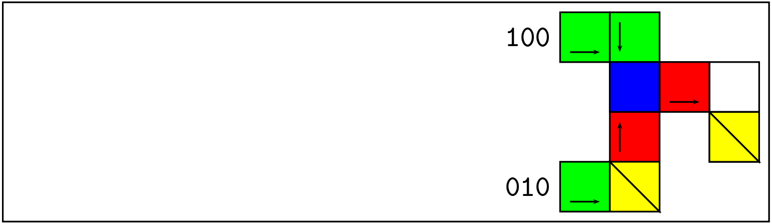

10000110
01000000

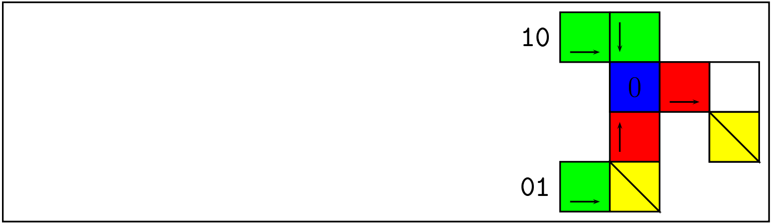

1000011
0
0100000

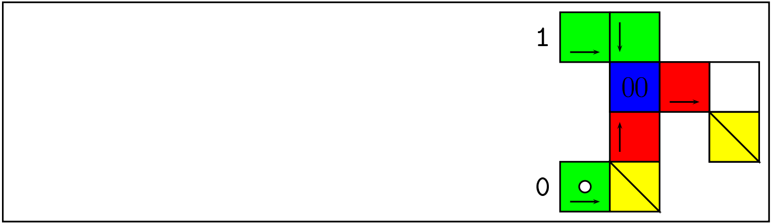

100001
10
010000

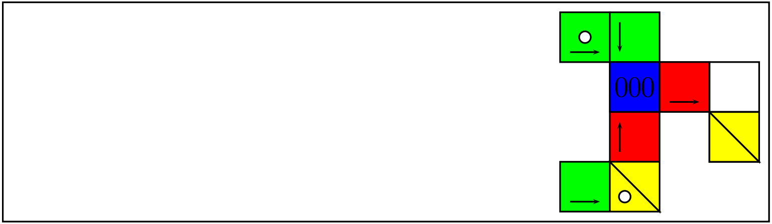

10000
01000

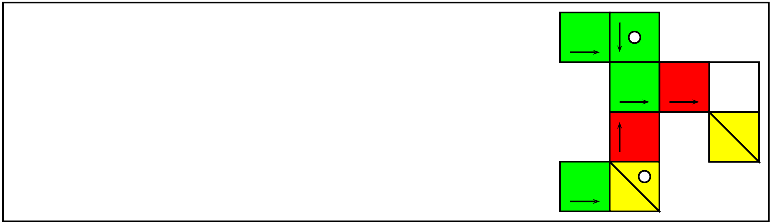

1000
0100

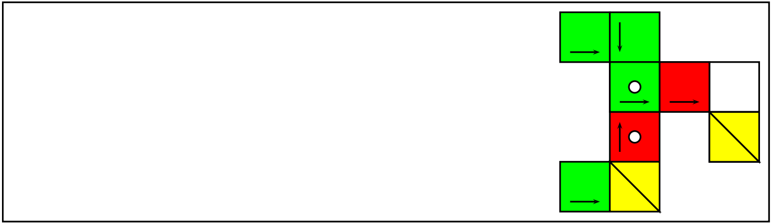

100
010

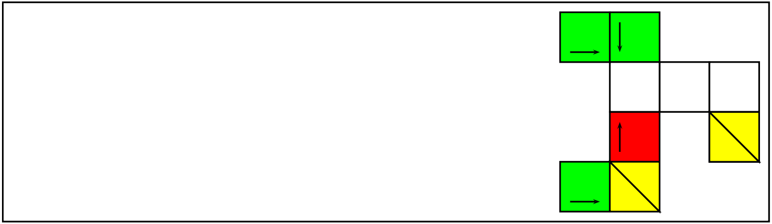

10
0
01


1
00
0


000







Fundamental questions reformulated:

(A) Logical universality. Can an automaton which performs the computations of a universal TM be embedded in von Neumann's 29-state CA? Yes!

Fundamental questions reformulated:

(A) Logical universality. Can an automaton which performs the computations of a universal TM be embedded in von Neumann's 29-state CA? Yes!

(B) Constructibility. Can an automaton be constructed by another automaton within von Neumann's 29-state CA? Yes!

Fundamental questions reformulated:

(A) Logical universality. Can an automaton which performs the computations of a universal TM be embedded in von Neumann's 29-state CA? Yes!

(B) Constructibility. Can an automaton be constructed by another automaton within von Neumann's 29-state CA? Yes!

(C) Construction-universality. Can a universal constructor be embedded in von Neumann's 29-state CA? Yes!

Fundamental questions reformulated:

(A) Logical universality. Can an automaton which performs the computations of a universal TM be embedded in von Neumann's 29-state CA? Yes!

(B) Constructibility. Can an automaton be constructed by another automaton within von Neumann's 29-state CA? Yes!

(C) Construction-universality. Can a universal constructor be embedded in von Neumann's 29-state CA? Yes!

(D) Self-reproduction. Can a self-reproducing automaton be embedded in von Neumann's 29-state CA? Yes!

Can there be embedded in von Neumann's 29-state CA an automaton which can perform the computations of a universal TM and can also reproduce itself? Yes!

Non-Trivial Self-Reproduction

Observation [Arthur Burks 1970]

This result is obviously substantial, but to express its real force we must formulate it in such a way that it cannot be trivialized.

Non-Trivial Self-Reproduction

Observation [Arthur Burks 1970]

This result is obviously substantial, but to express its real force we must formulate it in such a way that it cannot be trivialized.

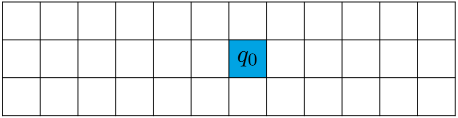
- Given a two-state cellular automaton where a state in the quiescent state changes into the other state if its neighbor to the north is non-quiescent.
- Then a single non-quiescent cell reproduces itself trivially in its neighbor.
- A requirement is needed that the self-reproducing automata have some minimal complexity.
- For example, requiring that the self-reproducing automata are also Turing-universal.

Herman's Cellular Automaton:

- Combine trivial, crystaline, self-reproduction with the finite control of a universal Turing machine.

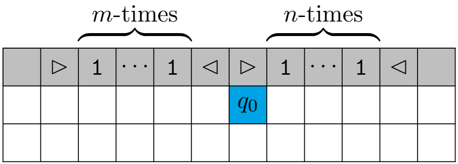
Herman's Cellular Automaton:

➜ Combine trivial, crystaline, self-reproduction with the finite control of a universal Turing machine.



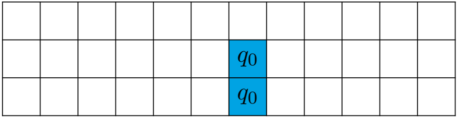
Herman's Cellular Automaton:

- Combine trivial, crystaline, self-reproduction with the finite control of a universal Turing machine.



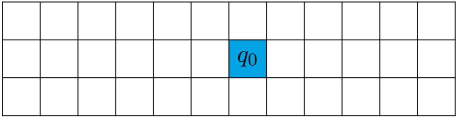
Herman's Cellular Automaton:

➜ Combine trivial, crystaline, self-reproduction with the finite control of a universal Turing machine.



Herman's Cellular Automaton:

- Combine trivial, crystaline, self-reproduction with the finite control of a universal Turing machine.



- What the result does show is that the existence of a self-reproducing universal computer in itself is not relevant to the problem of biological and machine self-reproduction.

Langton's Loop Cellular Automaton:

- Non-trivial self reproduction is characterised by separate processes of interpreting (translation) and copying (transcription) of a machine description.

Langton's Loop Cellular Automaton:

- Non-trivial self reproduction is characterised by separate processes of interpreting (translation) and copying (transcription) of a machine description.
- Basic idea: Sheath and core cells form a data path.

```
22222222222
11111111111
22222222222
```

Langton's Loop Cellular Automaton:

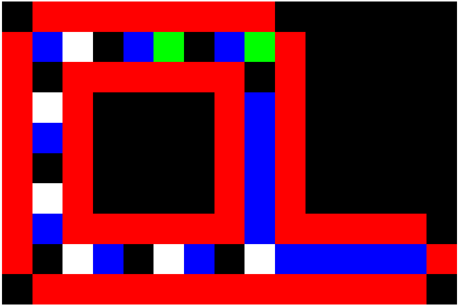
- Non-trivial self reproduction is characterised by separate processes of interpreting (translation) and copying (transcription) of a machine description.
- Basic idea: Sheath and core cells form a data path.

```
22222222222
11111111111
22222222222
```

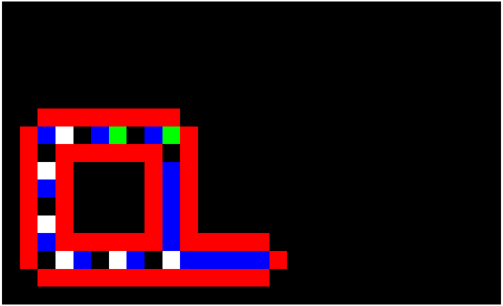
- Signals (genes) travel along the path and are copied at junctions.

```
     212
     212
     212
22222272222222
11061107111111
22222222222222
```

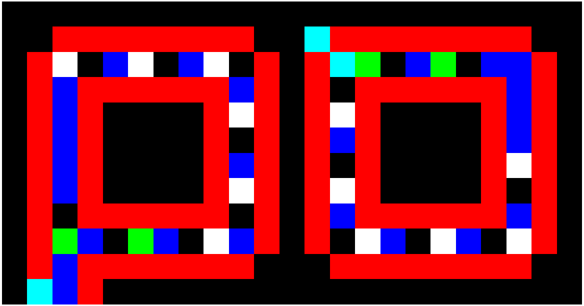
- At the end of the path a gene is interpreted to extend the path.



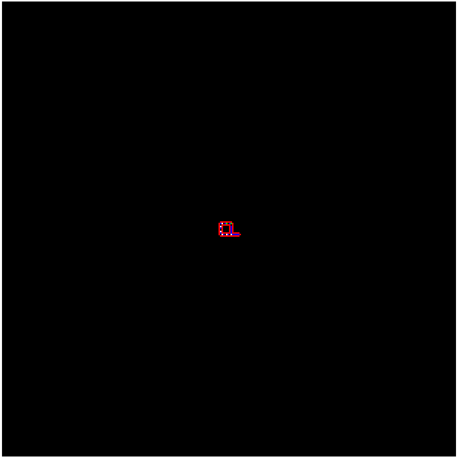
- The offspring is separated by the collision of genes at the new junction.



- If a loop tries to extend its arm to an occupied area, a sheath fragment is generated that absorbes all genes.



- Self-reproduction in infinite space.



So, what is non-trivial self-reproduction?

So, what is non-trivial self-reproduction?

John von Neumann 1949

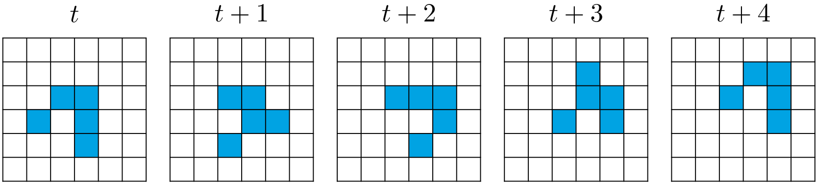
- ... A way around this difficulty [to define non-trivial self-reproduction] is to say that self-reproduction includes the ability to undergo inheritable mutations as well as the ability to make anothe organism like the original ...
- How can a machine manage to construct other machines more "complex" that themselves, in a general and open-ended way – that is, with the potential for unbounded evolutionary growth of complexity.

Game of Life

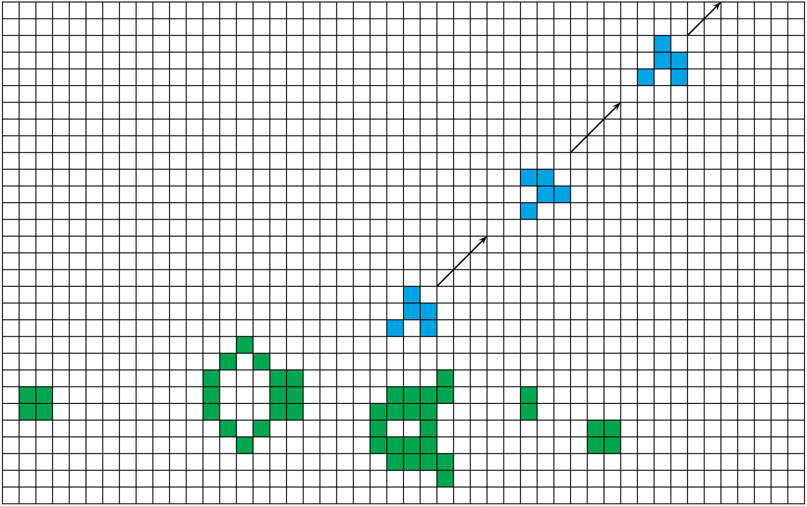
Example revisited:

- The state set is $\{0, 1\}$.
- Each cell is connected to its eight immediate neighbors.

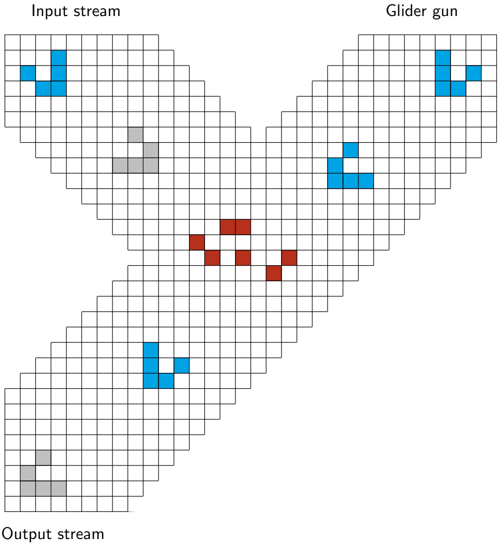
- A cell enters state 1, if the sum of their states is three.
- A cell keeps its current state, if the sum of their states is four.
- Otherwise the cell enters state 0.



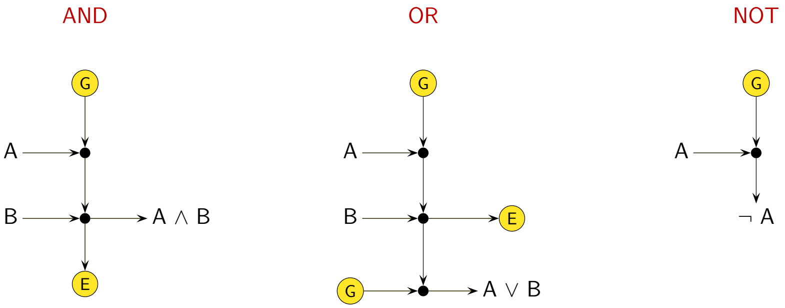
Game of Life: A glider.



Game of Life: A glider gun.



Game of Life: A NOT gate.



Game of Life: Logical gates.

Synchronization

The Firing Squad Synchronization Problem

Originally, the problem has been stated as follows:

> Consider a finite but arbitrary long chain of finite automata that are all identical except for the automata at the ends.

Synchronization

The Firing Squad Synchronization Problem

Originally, the problem has been stated as follows:

> Consider a finite but arbitrary long chain of finite automata that are all identical except for the automata at the ends. The automata are called soldiers, and the automaton at the left end is the general.

Synchronization

The Firing Squad Synchronization Problem

Originally, the problem has been stated as follows:

> Consider a finite but arbitrary long chain of finite automata that are all identical except for the automata at the ends. The automata are called soldiers, and the automaton at the left end is the general. The automata work synchronously, and the state of each automaton at time step $t+1$ depends on its own state and on the states of its both immediate neighbors at time step t.

Synchronization

The Firing Squad Synchronization Problem

Originally, the problem has been stated as follows:

> Consider a finite but arbitrary long chain of finite automata that are all identical except for the automata at the ends. The automata are called soldiers, and the automaton at the left end is the general. The automata work synchronously, and the state of each automaton at time step $t + 1$ depends on its own state and on the states of its both immediate neighbors at time step t. The problem is to find states and state transitions such that the general may initiate a synchronization in such a way that all soldiers enter a distinguished state, the firing state, for the first time at the same time step.

Synchronization

The Firing Squad Synchronization Problem

Originally, the problem has been stated as follows:

> Consider a finite but arbitrary long chain of finite automata that are all identical except for the automata at the ends. The automata are called soldiers, and the automaton at the left end is the general. The automata work synchronously, and the state of each automaton at time step $t+1$ depends on its own state and on the states of its both immediate neighbors at time step t. The problem is to find states and state transitions such that the general may initiate a synchronization in such a way that all soldiers enter a distinguished state, the firing state, for the first time at the same time step. At the beginning all non-general soldiers are in the quiescent state.

Theorem

The minimal solution time for the FSSP is $2n - 2$, where n is the length of the array (chain).

Algorithm

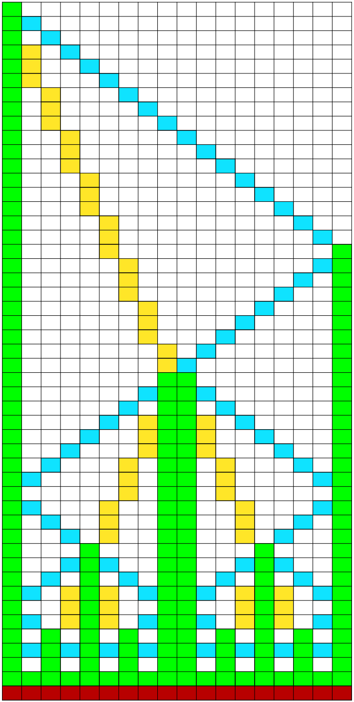
- The problem can be solved by dividing the array in two, four, eight etc. parts of (almost) the same length until all cells are cut-points.
- Exactly at this time the cells will fire synchronously.

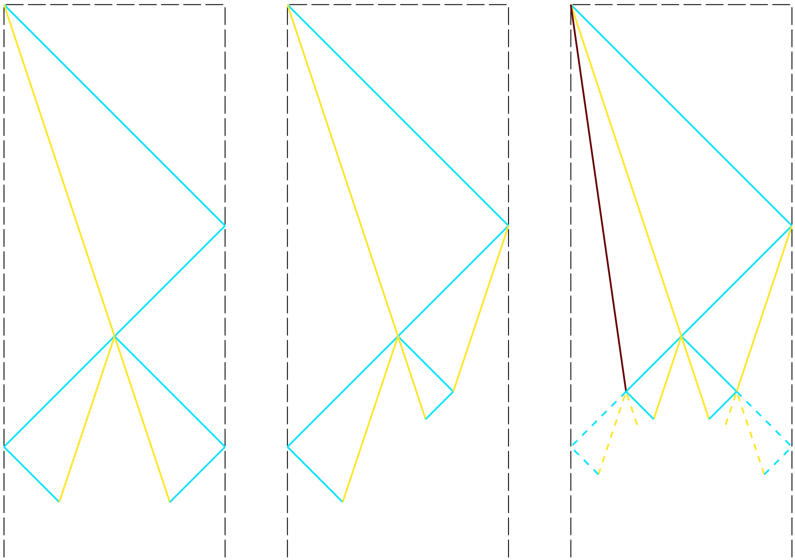
Algorithm

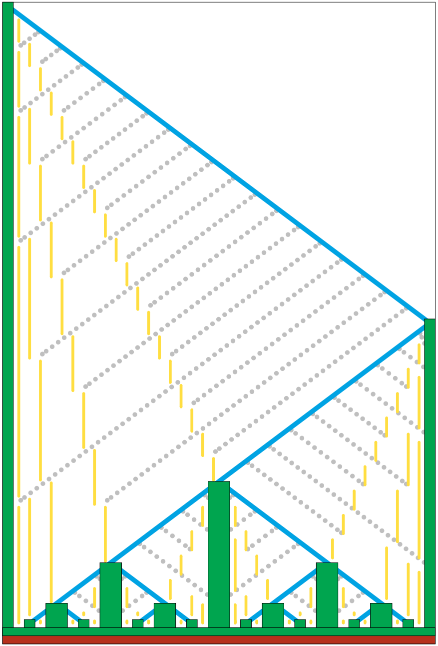
- The problem can be solved by dividing the array in two, four, eight etc. parts of (almost) the same length until all cells are cut-points.
- Exactly at this time the cells will fire synchronously.
- The divisions are performed recursively. At first the array is divided into two parts. Then the process is applied to both parts in parallel, etc.

Algorithm

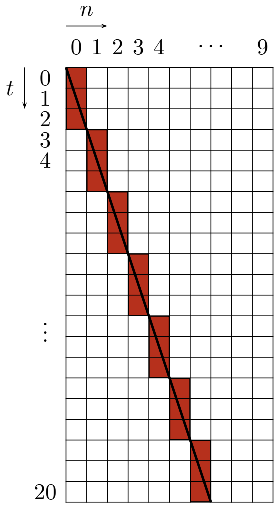
- The problem can be solved by dividing the array in two, four, eight etc. parts of (almost) the same length until all cells are cut-points.
- Exactly at this time the cells will fire synchronously.
- The divisions are performed recursively. At first the array is divided into two parts. Then the process is applied to both parts in parallel, etc.
- In order to divide the array into two parts, the general sends two signals $S1$ and $S2$ to the right.
- Signal $S1$ moves with speed one, and signal $S2$ with speed $1/3$.



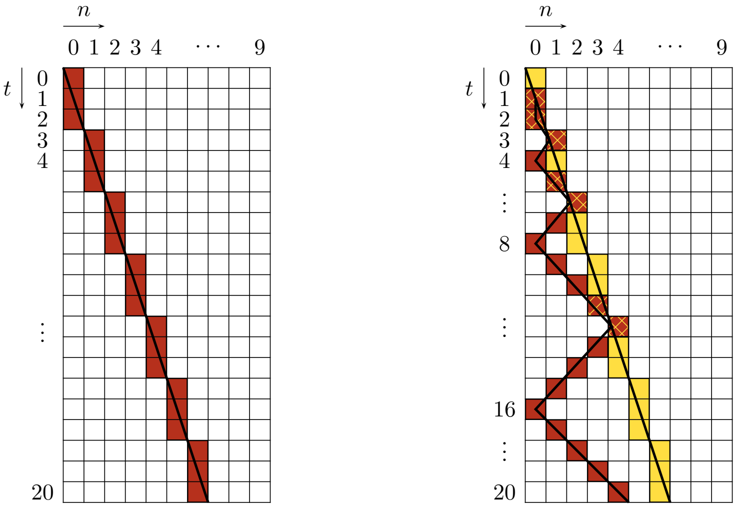




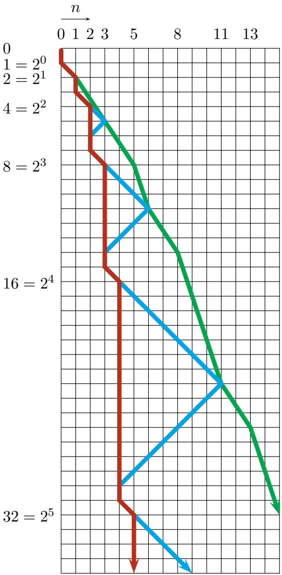
Signals

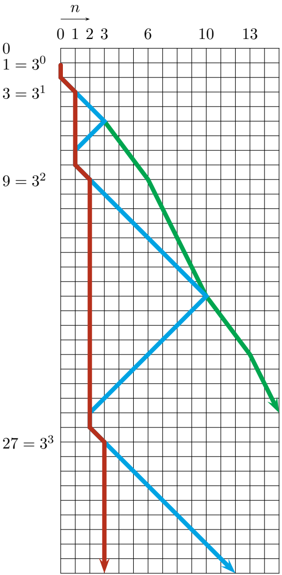


Signals

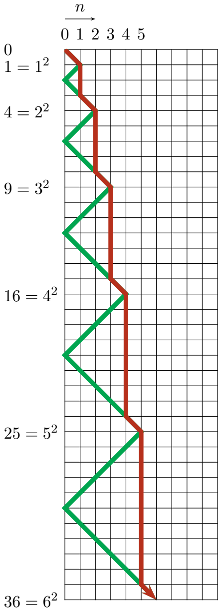


Signals with exponential characteristic function b^n, $b \geq 2$





Signals with polynomial characteristic function n^b, $b \geq 1$



- A signal with characteristic function n^2 can be derived from $(n+1)^2 = n^2 + 2n + 1$.
- In particular, signal ξ has to stay for $2n$ time steps in cell n, and subsequently has to move in one step to cell $n+1$.
- The delay is exactly the time needed by an auxiliary signal that moves from cell n to cell 0 and back.

Signals with polynomial characteristic function n^b, $b \geq 1$

- It holds $(n+1)^3 = n^3 + 3n^2 + 3n + 1$.
- Therefore, a signal with characteristic function n^3 has to stay for $3n^2 + 3n$ time steps in cell n and subsequently has to move in one step to cell $n+1$.
- The delay $3n$ is exactly the time needed by an auxiliary signal that moves from cell n to cell 0 and back and once more to cell 0.
- In cell 0 a modified quadratic signal is generated, which moves from cell 0 to cell n and back and once more to cell n.