Evolving Structures in Mathematics

Seminar at Charles University, Prague

Tomas Mikolov, 2019

Today’s talk

Evolution should allow increasingly more complex patterns to emerge.

What is complex, and what is simple?

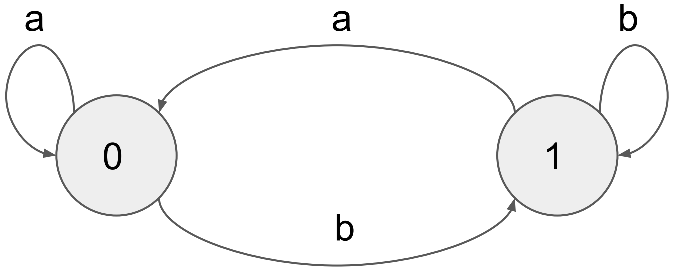
Complexity measures: The Quark and The Jaguar, Murray Gell-Mann

Introduction to cellular automatons.

Finite state machines

State is a discrete finite number.

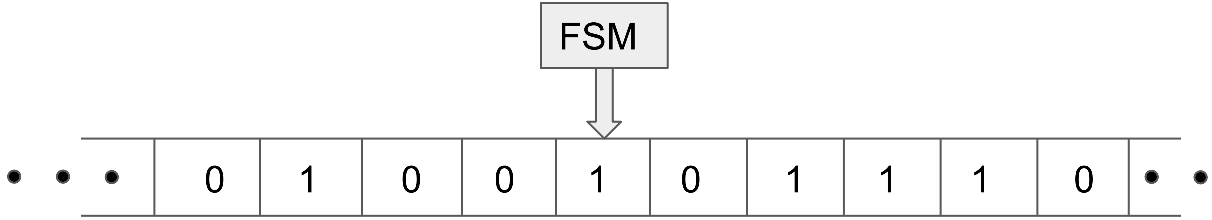
Transition function defines the next state given current state and an input symbol.



Turing Machine

Finite machine operating on an unbounded tape.

Turing machine can represent any algorithm.



Turing-completeness

There are many computational models as powerful as a Turing machine.

Some Turing-complete models are even simpler than Turing machine.

Computational complexity of algorithms can vary based on the machine.

Halting problem

There exists no algorithm that can decide if some Turing machine will reach a specific state.

Certain algorithms assume that programs eventually reach some final state, but this cannot be decided.

Minimum description length

MDL(S) = minimum length of a program that generates a sequence S

MDL(“`abababababababababab`”) < MDL(“`aabaaabbaaabaaaabab`”)

MDL cannot be computed.

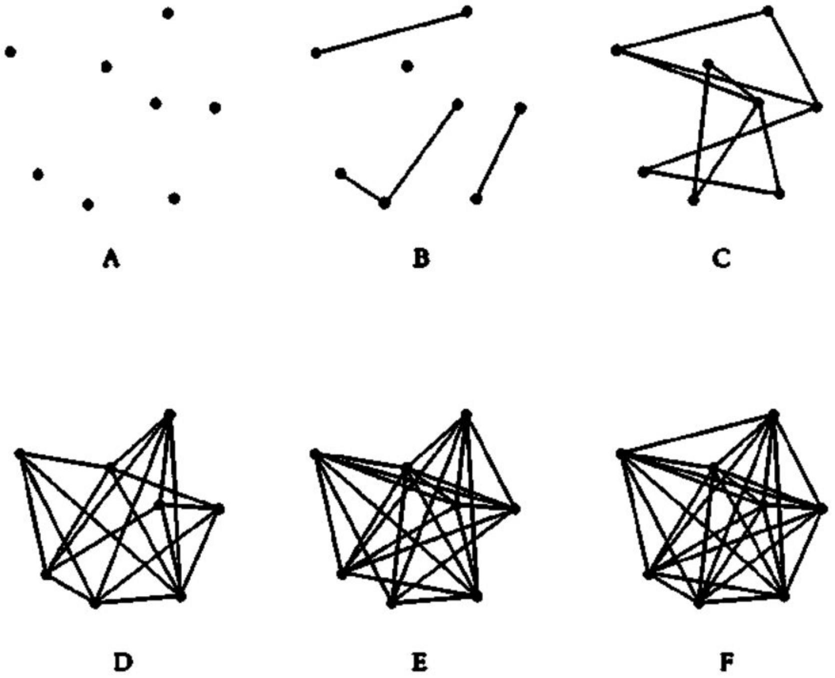
Algorithmic probability

Probability that a random computer program will generate a sequence S.

Considers all programs that can generate S, but is dominated by the shortest ones.

We can construct optimal predictor by considering all programs that generate S and at least one more symbol.

What is complex, and what is simple?



Some patterns of connection of eight dots.

What is complex, and what is simple?

Complex = high description length? Random strings are not complex…

Complex = a lot of structure? How do we measure it?

Gell-Mann: MDL is a sum of regularities, and incompressible noise. The number of regularities is then the effective complexity.

Cellular automatons

Regular grids of automatons that perform computation in parallel based on their state and state of their neighbors.

Even very simple cellular automatons can exhibit complex behavior.

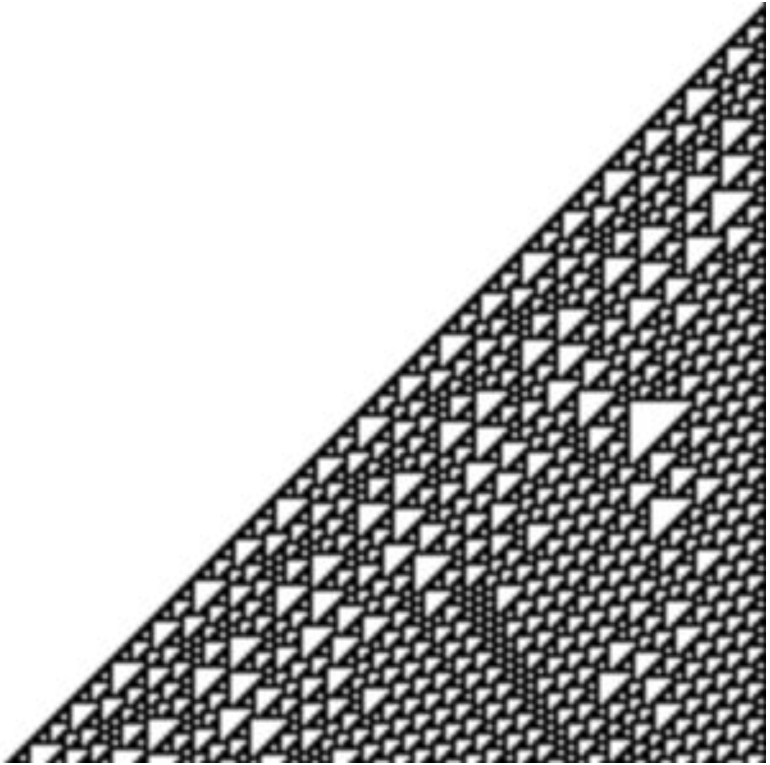
1D cellular automatons

Most basic cellular automatons: 1D tape, binary states, the next state depends on the immediate neighbors: total ~256 automatons.

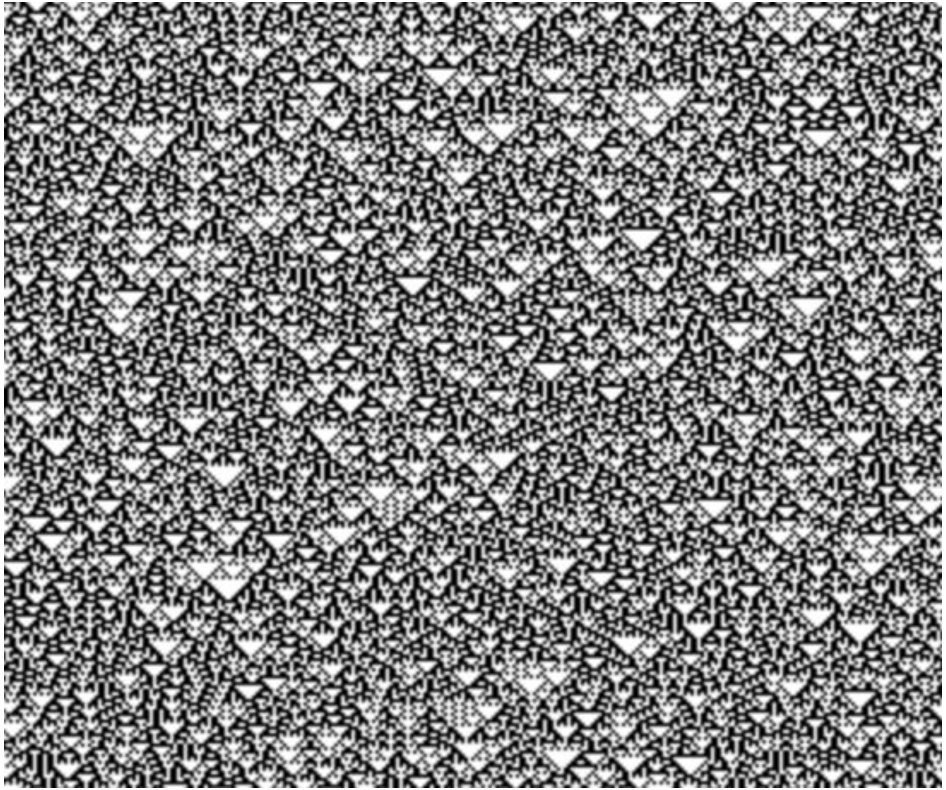
Example: ‘Rule 110’ (it’s Turing-complete!):

Current pattern	111	110	101	100	011	010	001	000
New state for center cell	0	1	1	0	1	1	1	0

Rule 110



Rule 90



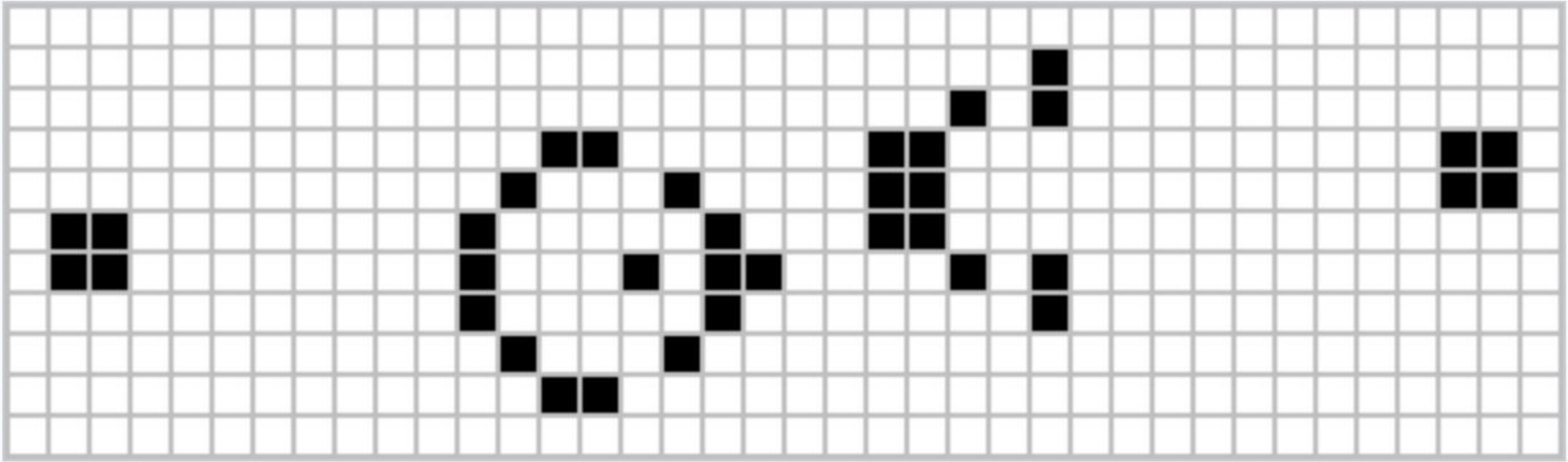
Conway’s Game of Life

2D cellular automaton with hand-designed transition rules.

Turing-complete: can perform any algorithm if certain initial state is provided.

Many interesting emergent properties: structures that move across space, oscillating behavior etc.

Conway’s Game of Life



Gosper glider gun