Timing and stability in cellular automata

Konstantin Klemm

Bioinformatics Group
University of Leipzig, Germany

Definition

- finite set of states S, often $|S| = 2$
- mapping ("rule table") $f : S^z \to S$
- a lattice of n sites with coordination number z where site i has neighbours $a(i,1), a(i,2), \ldots a(i,z)$
- time-discrete dynamics of lattice site i

$$s_i(t+1) = f[s_{a(i,1)}(t), \ldots s_{a(i,z)}(t)]$$

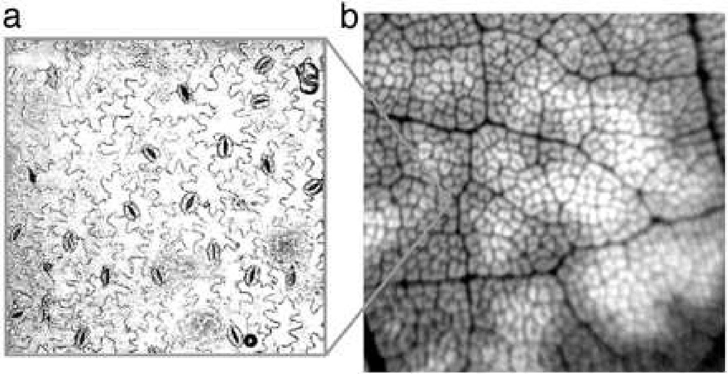
All lattice sites are updated in synchrony.

Purpose

- Computer Science: Models of computation,
  e.g. "Game of Life" and "rule 110" are Turing-complete.
- Artificial Life: Study of self-reproducing structures
- Physics: Abstractions of spatio-temporal dynamics, pattern formation

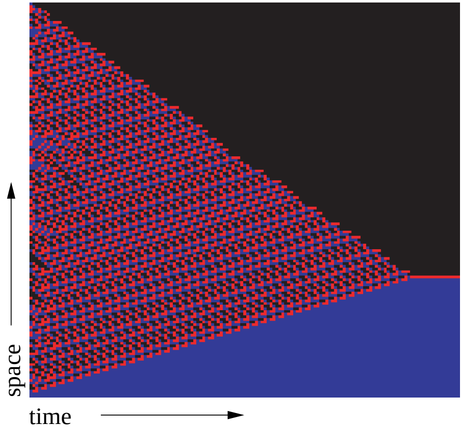
Motto: Simplest rules may yield most complex patterns / computations.

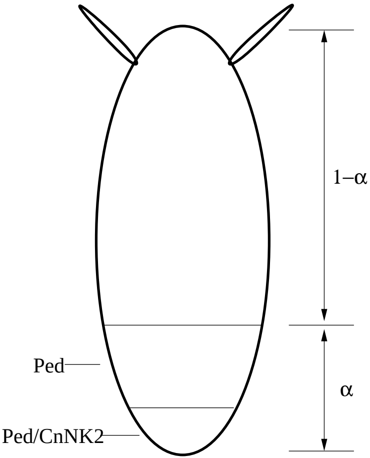
Putative CA dynamics on plant leaves



Peak et al., PNAS (2004)

CA model for morphogenesis in Hydra

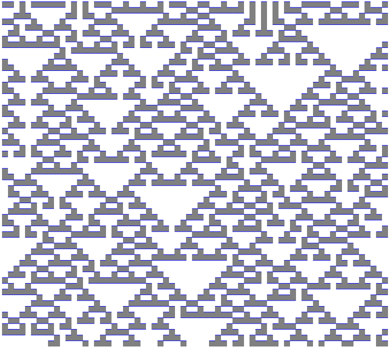




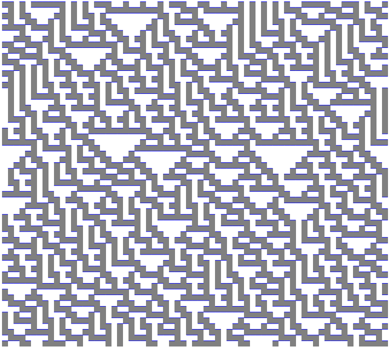
Rohlf & Bornholdt, JSTAT (2005)

Seashell

Seashell and elementary CA rule 22



Seashell and elementary CA rule 30



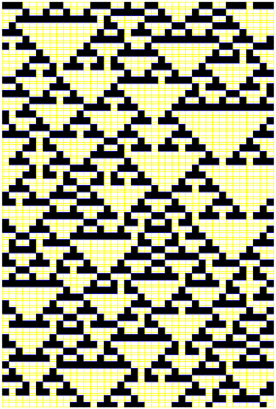
Role of noise?

- Are the observed patterns robust under small stochastic perturbations?
- Stability analysis in continuous dynamical systems: Make a small perturbation and check if the system returns to the fixed point / limit cycle.
- Here: Discretized state space. What is a "small" perturbation?
- "Typical" implementation of noise in CA:

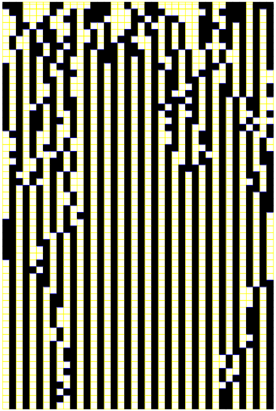
Stochastic asynchronous update

Rule 22, comparison sync / async update

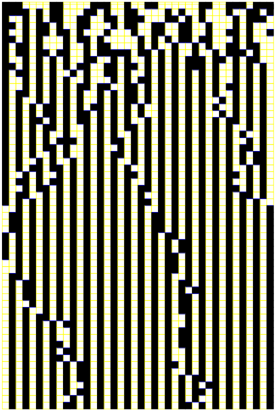
synchronous update



asynchronous update

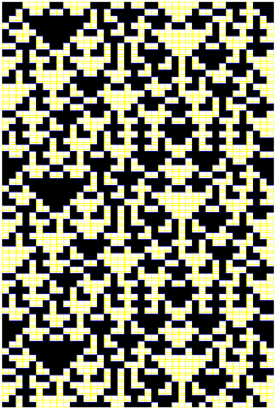


async, another realization



Rule 150, comparison sync / async update

synchronous update

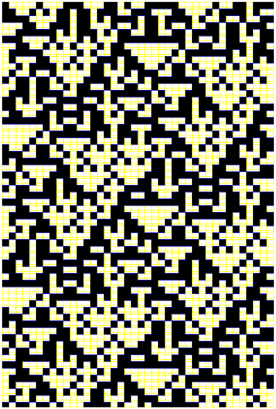


asynchronous update

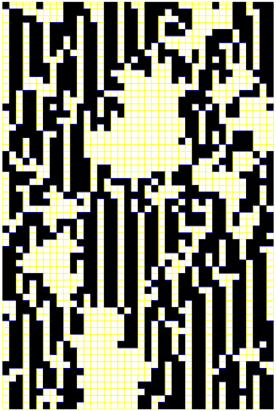
async, another realization

Rule 90, comparison sync / async update

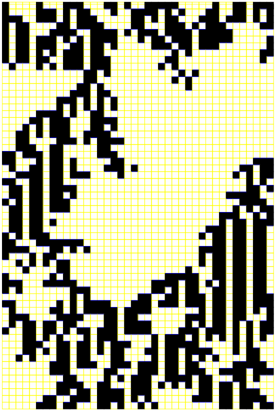
synchronous update



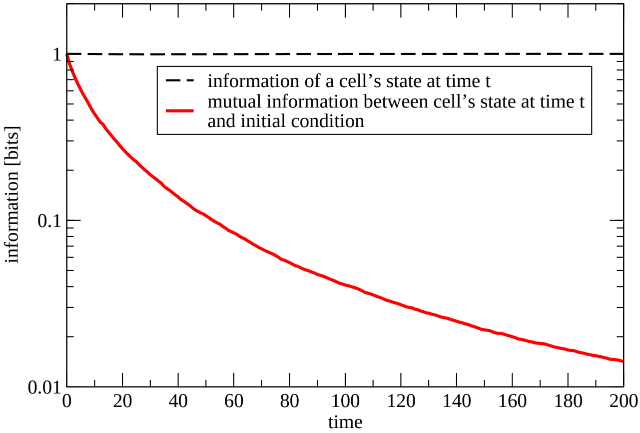
asynchronous update



async, another realization



Loss of memory



Evolution of mutual information between one cell's state at time t and the full initial configuration.

Summary so far

- Deterministic CA:
  "Complex" (interesting) spatio-temporal pattern formation
- CA with noise implemented as stochastic update order:
  Largely irreproducible dynamics
- Devastating effect of stochastic asynchronous update known for long, cf. Ingerson and Buvel, Physica D (1984)

How to proceed

Solution 0

Use more states and more complicated interactions to implement additional clock signals
cf. Nehaniv, Int. J. Alg. Comp. (2004)

Solution 1

- Stick to simple rules
- Consider a less destructive type of perturbation.
- Get happy.

Framework for perturbed timing

- Now time $t \in \mathbb{R}$, no longer discrete.
- Expand initial condition $s(0)$ to unit interval:

$$s_i(t) = \begin{cases} f_i(s(0)) & , \quad \text{if } t > 1 - \delta_i \\ s_i(0) & , \quad \text{otherwise} \end{cases}$$

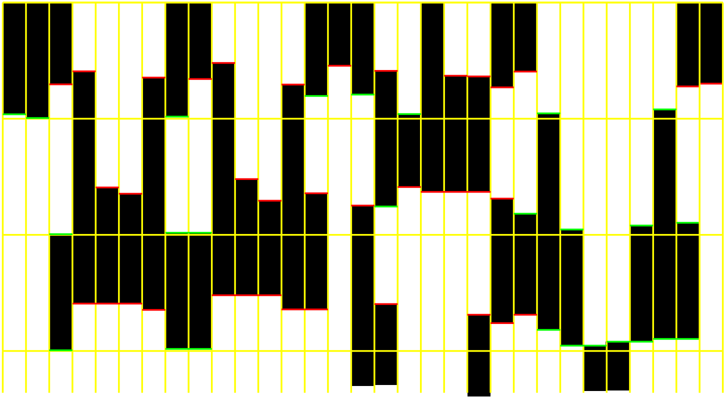
  where $\delta_i \in [0, \epsilon]$ is drawn randomly and independently for each node i.
- Let the system evolve according to

$$s_i(t+1) = \Theta \left[(2c)^{-1} \int_{t-c}^{t+c} f_i(s(\tau)) \mathrm{d}\tau - 1/2 \right]$$

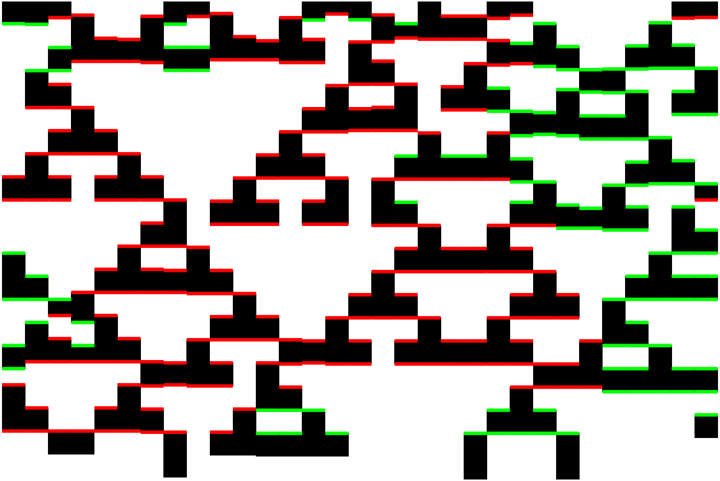
  where Θ is the step function and $1/2 > s \geq \epsilon > 0$.
- Check, if system autonomously regains synchrony.

Klemm & Bornholdt, PNAS 102, 18414 (2005); Phys. Rev. E 72, 055101(R) (2005).

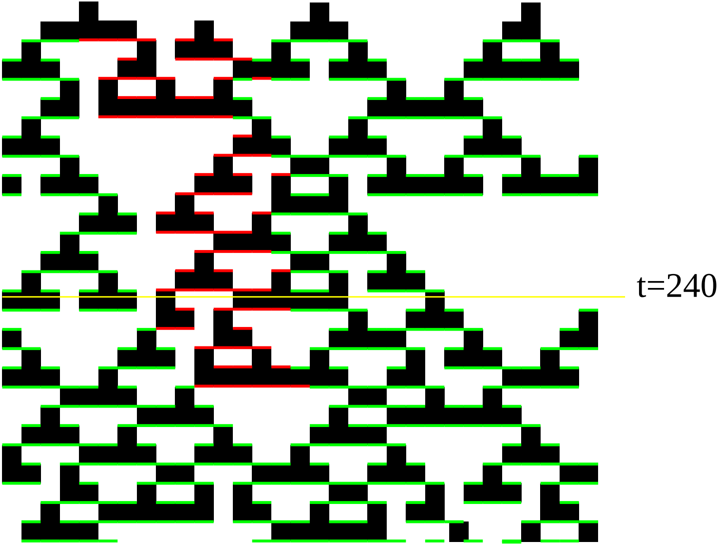
CA rule 22, initial timing perturbation



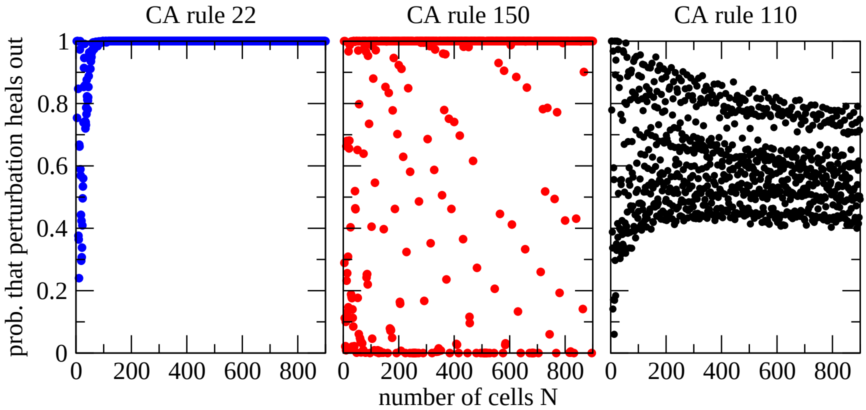
CA rule 22, spatial coarsening of time lags



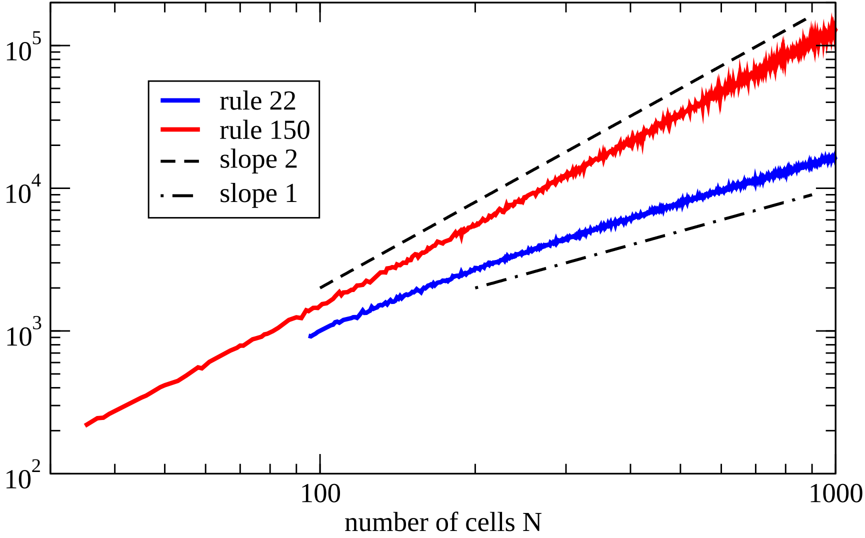
CA rule 22, timing perturbation heals



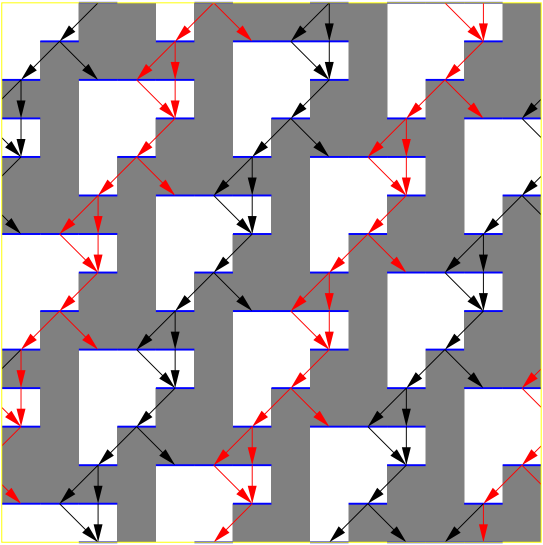
Probability of healing



Time until healing



Unstable pattern in rule 110



Overview of results

- stable elementary rules
  0, 4, 22, 32, 36, 54, 72, 76, 104, 128, 132, 160, 164, 200, 204, 218, 222, 236, 250,254
- partially stable el. rules (strong dependence on n)
  90, 122, 126, 150
- unstable elementary rules
  50, 94, 108, 110, 178
- Elementary CA fail to produce sustained synchronous blinking of all cells
- Game of Life: Blinkers, gliders, spaceships etc. are unstable

Further results

Summary

Stable and unstable CA rules can be distinguished by their response to minimal timing perturbations.