Preconditions

--

Objectives

Example-based introduction to Organic Computing

Content

- Examples
  - Sorting network
  - Indian junction
  - Predator - prey
  - Balinese water temples
  - Dissipative structures
  - Candle mover and ants
  - Tierra und self-modifying code
  - Small World
  - Cellular Automaton
  - Lindenmayer Systems
- Common characteristics
- Technical usage: Organic Computing

© C. Müller-Schloer 2015

Sven Brückner‘s sorting network

?

© C. Müller-Schloer 2015

Sven Brückner‘s sorting network

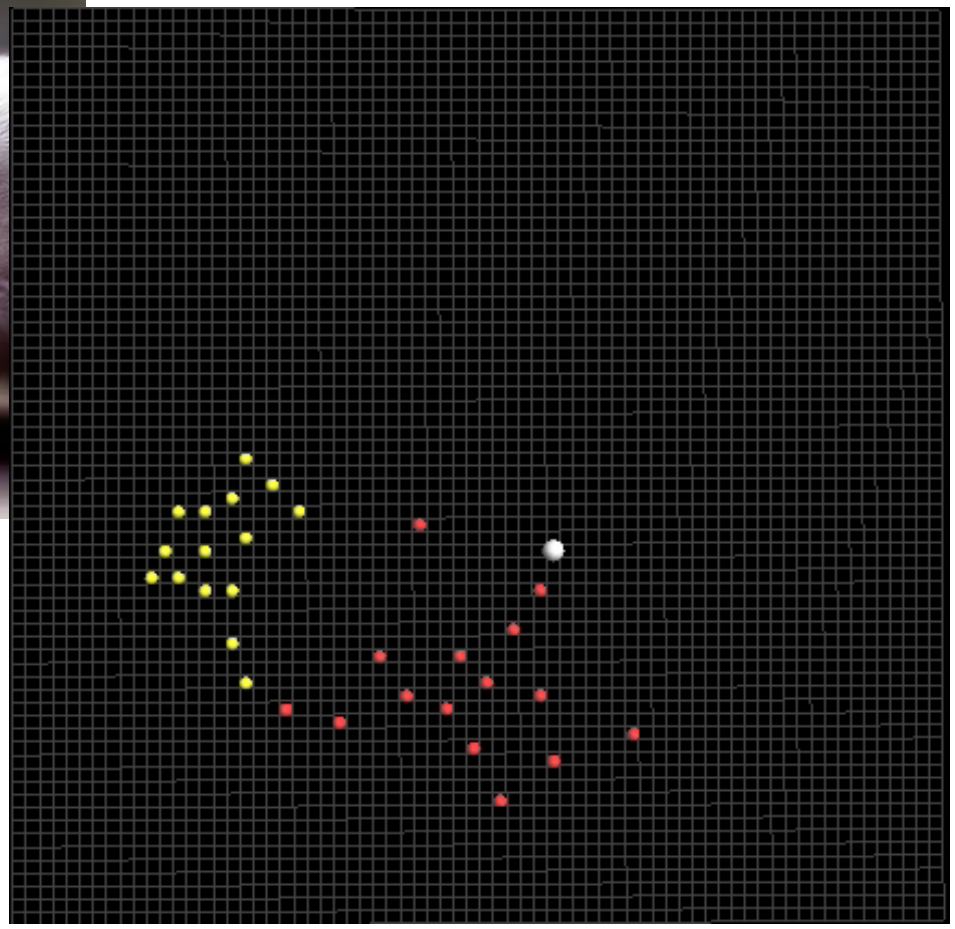
1. Move same color to output as last time.
2. If not possible: sleep (with probability p)
3. Move a random part.

© C. Müller-Schloer 2015

Indian junction

© C. Müller-Schloer 2015

Predator - prey



© C. Müller-Schloer 2015

Balinese water temples

© C. Müller-Schloer 2015

Bali: Terraces for rice production

© C. Müller-Schloer 2015

Balinese water temples (1)

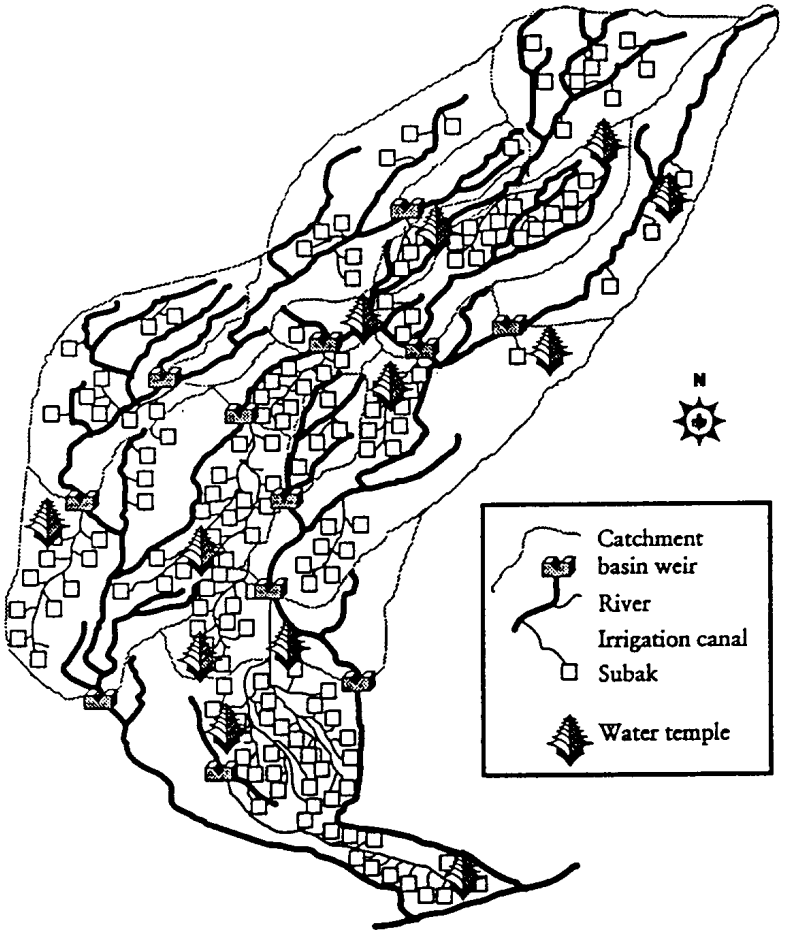
- Watering system for the cultivation of rice
  - Main factors:
    - Water circulation
    - Alternation between dry and wet periods
  - Objectives:
    - PH-value
    - Activity of micro-organisms
    - Distribution of mineral nutrient
    - Herbicide
    - Pest control (for large areas)
    - Stabilisation of temperature
  - Problem: Synchronous watering leads to peak demand of water!

See: Lansing, Kremer

© C. Müller-Schloer 2015

Balinese water temples (2)

- Problem for each farmer
  - Determine cultivation sequence
- Target
  - Maximise crop
- Attributes of cultivation sequence
  - Phases of cropping (date)
  - Cultivar (kind of plant)
  - Watering
  - Drying



The Oos and Petanu rivers in south-central Bali (not to scale).[16]

Subak: agricultural association

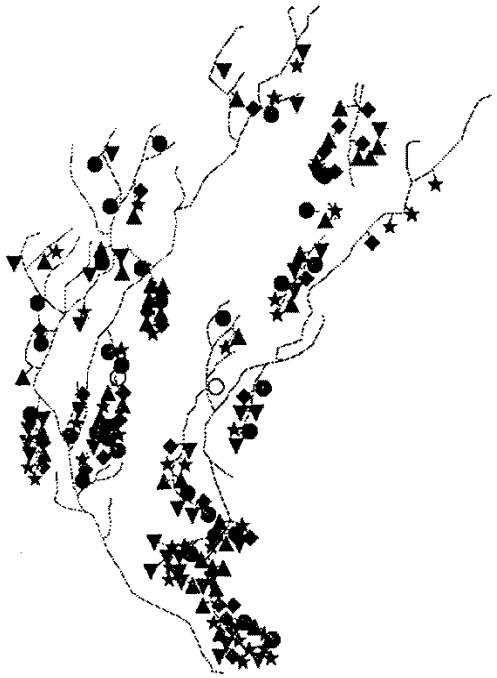
© C. Müller-Schloer 2015

Balinese water temples (3)

- How to determine the optimal sequence?
  - Trial and error vs. planning
  - Coordination: global or local?
  - Is the solution suitable for the local problem?
  - Is the solution adaptable?
- Hypothesis: Coordination algorithm
  => synchronous, local, like the best neighbour (co-adaptation)
- Process: Simulation
  => Crop is modelled as a function of cultivation sequence and environment
- Start: Randomised initialisation

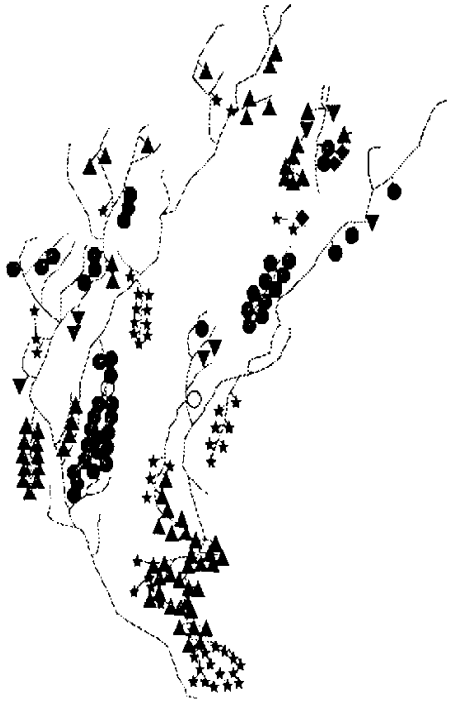
© C. Müller-Schloer 2015

❑ Cultivation sequences



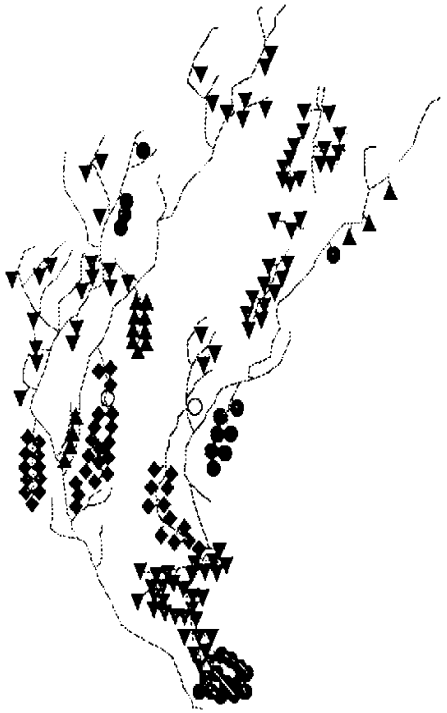
Randomised distribution at start-up

4.9 tons/ha



Simulated pattern after co-adaptation cycles

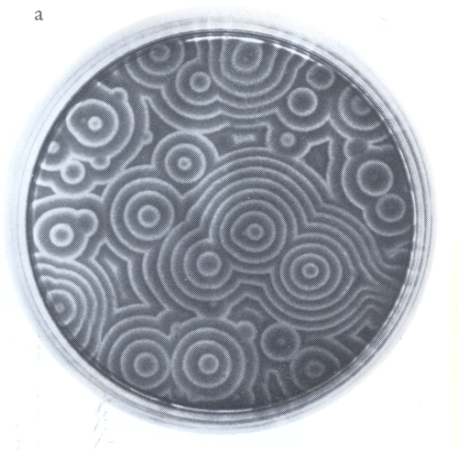
8.6 tons/ha

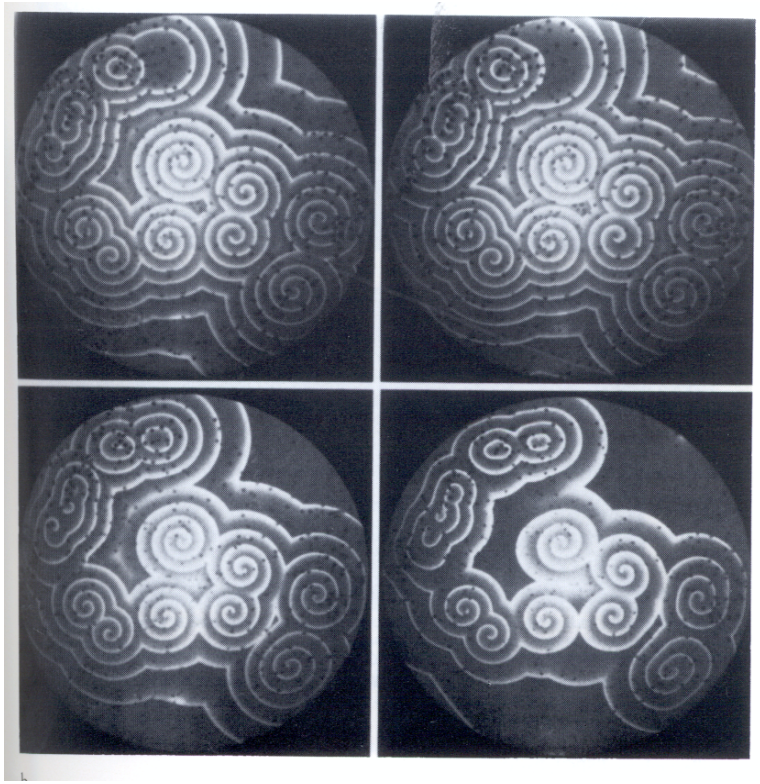


Traditional system of cave temples at Bali

© C. Müller-Schloer 2015

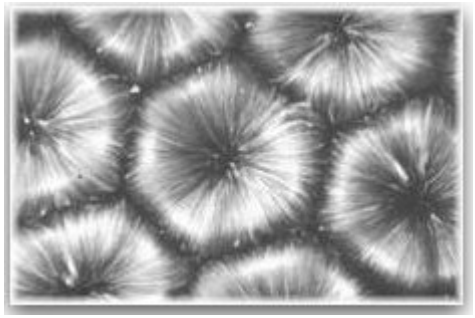
- Ilya Prigogine: Belousov-Zhabotinsky-Reaction
  - Auto-catalytic reaction
  - Space-time pattern
  - Equilibrium of flow
  - „Living“ until energy is consumed





© C. Müller-Schloer 2015

More Dissipative Structures



- Benard cell: A type of **dissipative structure**, i.e. heat dissipation through the liquid, works to form hexagonal structures.
- A **dissipative structure** is an organized nonequilibrium state of matter created and maintained due to dissipative processes. [1] The term was proposed in 1967 by Belgian chemist and thermodynamicist Ilya Prigogine to describe the spontaneous appearance of ordered structures in the non-linear domain, far from equilibrium. [2] Classic examples of dissipative structures include: Bénard cells (adjacent), the Belousov-Zhabotinskii reaction, Taylor vortices, cyclones, hurricanes, and lasers. [3] The loose spin-off of the theory is that all life forms, humans included, are, by definition, far-from-equilibrium dissipative structures. [4] A newer variant of the term is "dissipative system".
- http://www.eoht.info/page/Dissipative+structure

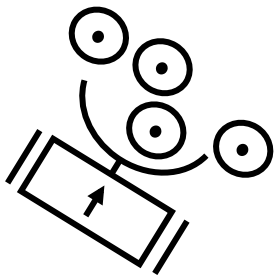
© C. Müller-Schloer 2015

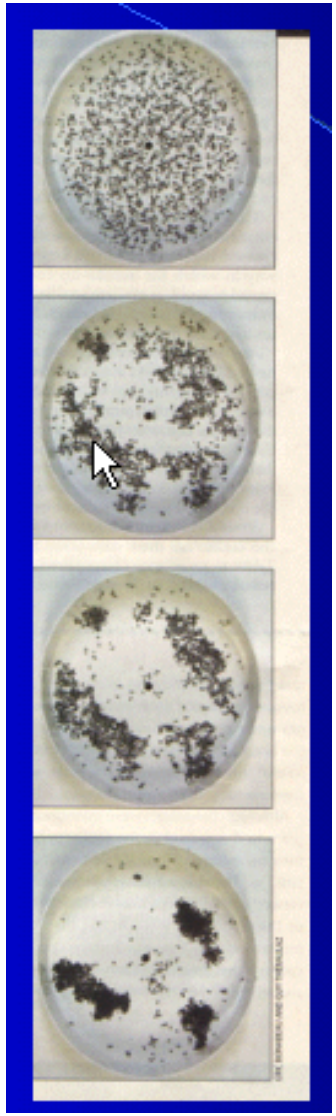
Candle Mover

1 Sensor (pressure)

1 Actuator (direction)

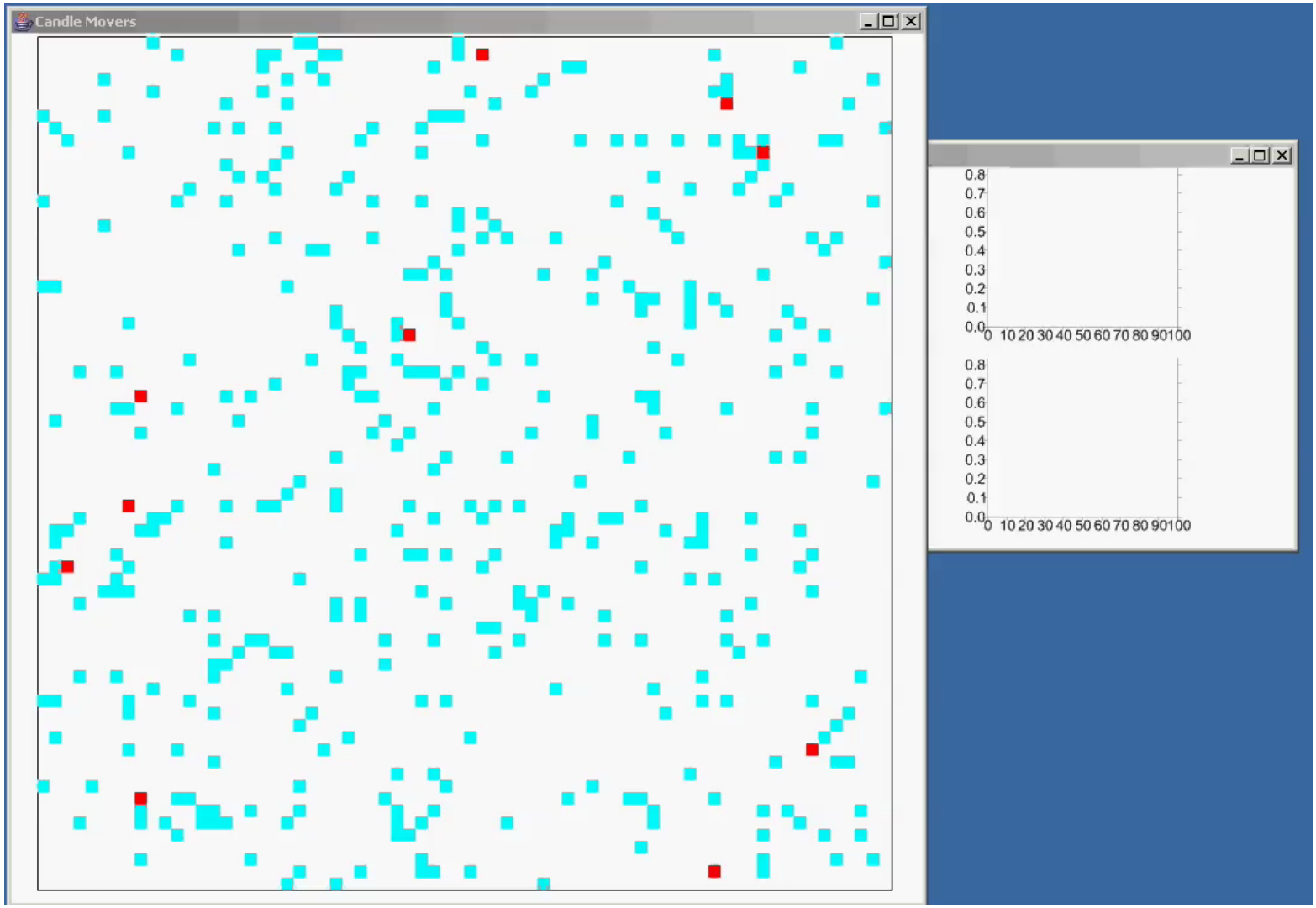
> 2 candles: move backwards, turn, move forwards





© C. Müller-Schloer 2015

Candle mover



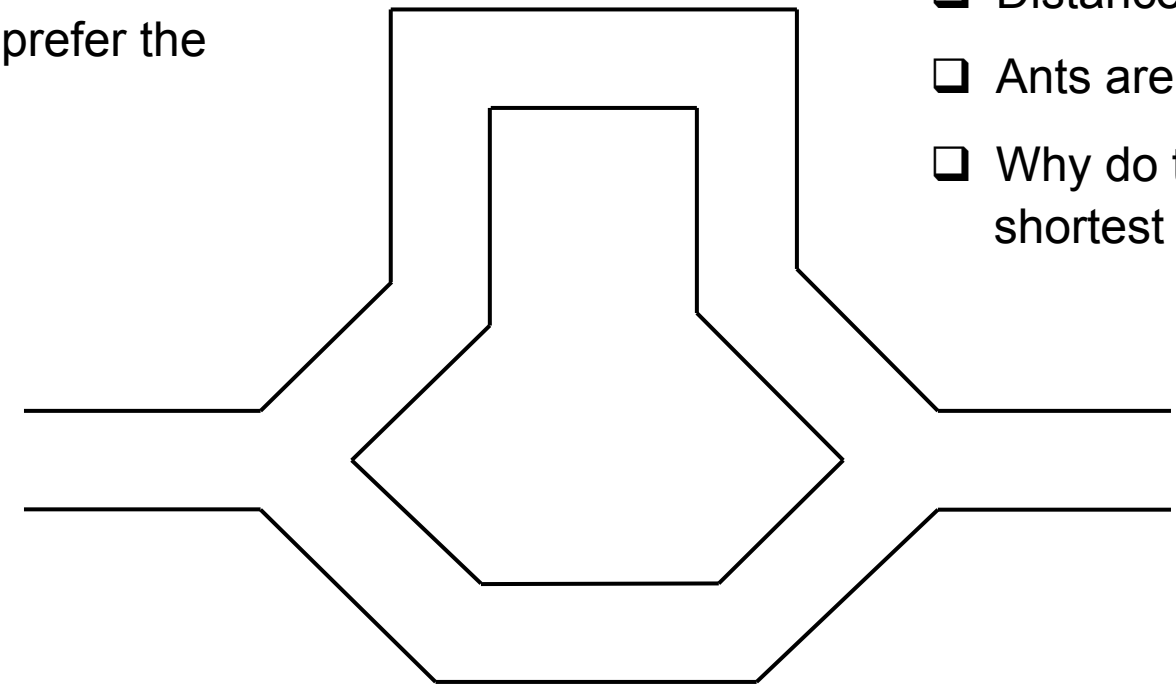
© C. Müller-Schloer 2015

Behaviour of ants …

(Deneubourg et al., Dorigo, around 1990)

- Ants explore both paths
- They find and prefer the shorter path.

Nest



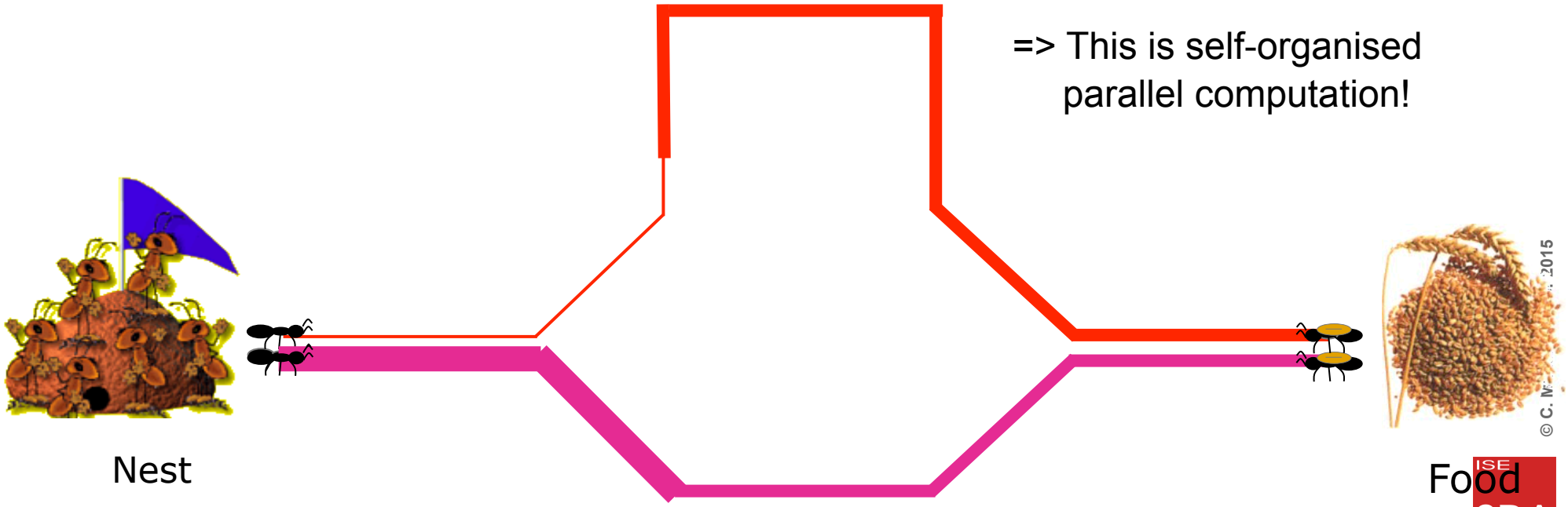
- Distances are unknown
- Ants are short sighted.
- Why do they find the shortest path?

Food

© C. Müller-S...

- ❑ Ants deposit pheromones on their path.
- ❑ Ants prefer paths having higher pheromone concentration.
- ❑ Pheromones evaporate.

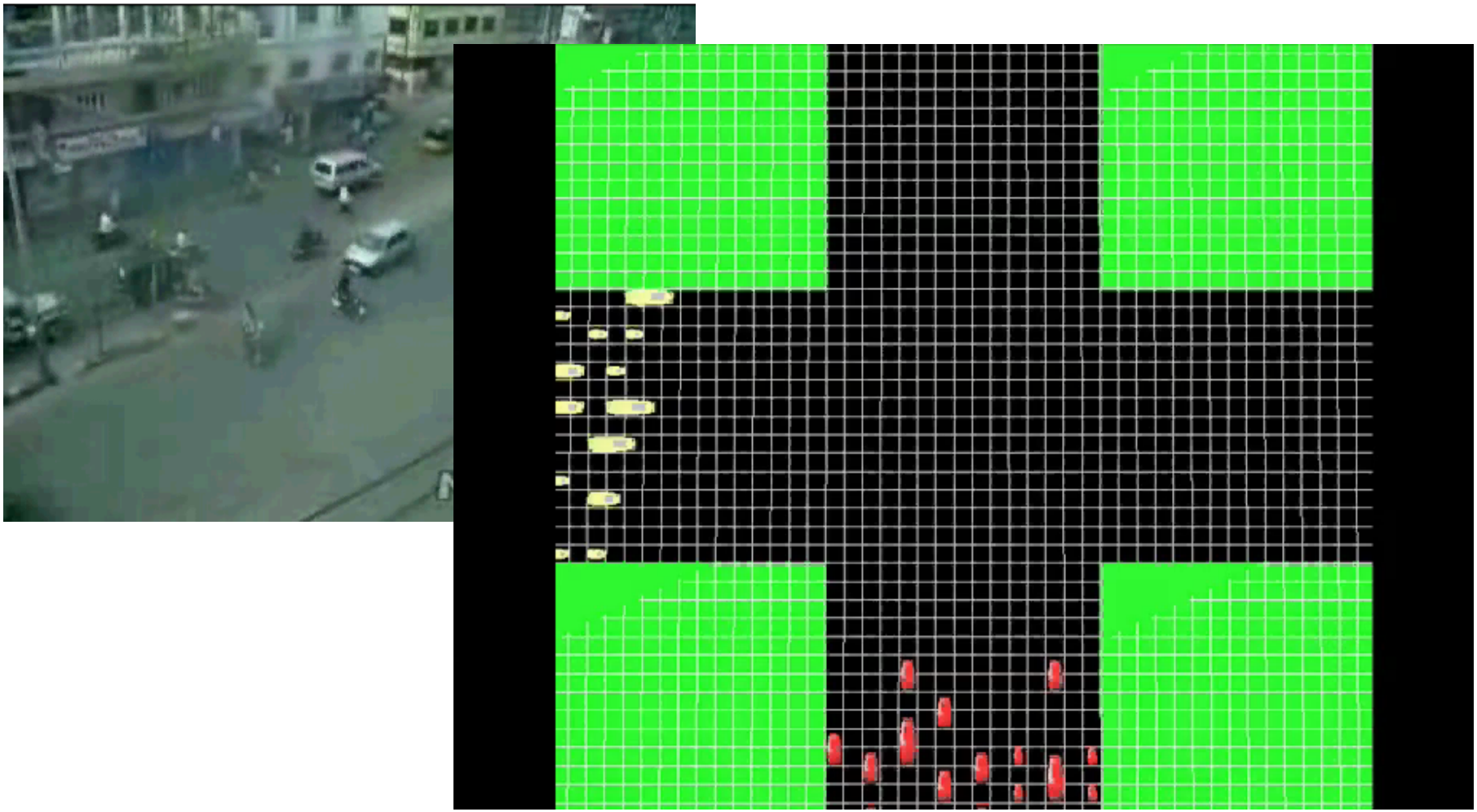
→ Ants don‘t care about shortest paths but only about not getting lost and finding food.



Negative Emergent Behavior of ants

© C. Müller-Schloer 2015

- „Indian junction“ vs. self-organized junction

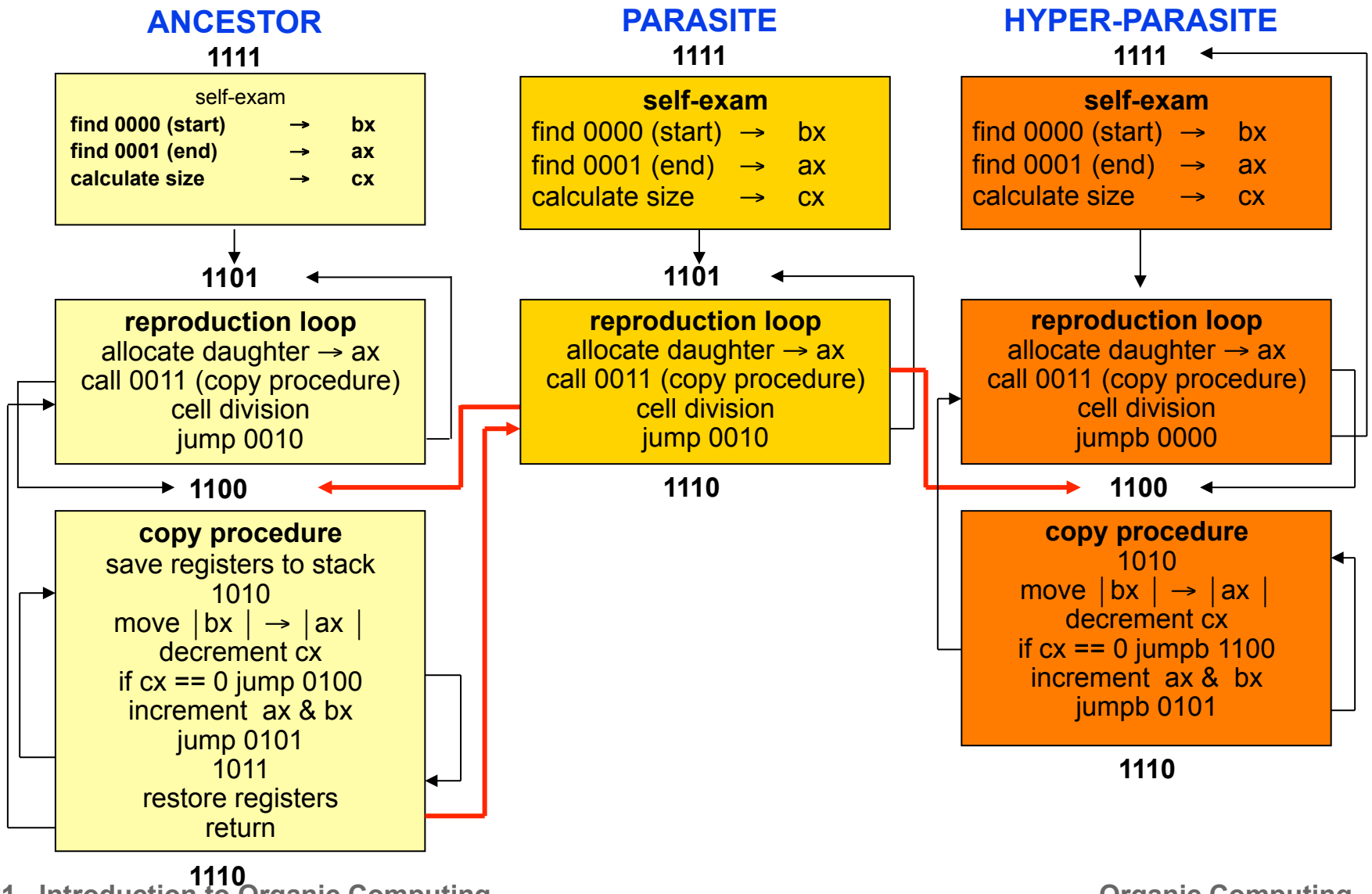


© C. Müller-Schloer 2015

- Tierra
  - Developed by Thomas S. Ray (http://life.ou.edu/tierra/)
  - Individuals (= program) compete against each other:
    - Memory
    - CPU time

- Programming language as „genome“
- Biotope (living space): 60 kByte (soup)
- Sole capability: self-reproduction
- Errors during copy process: mutation
- Mortality: Increased probability due to errors
- CPU time is allocated depending on size (small individuals are more often processed than bigger ones)

© C. Müller-Schloer 2015

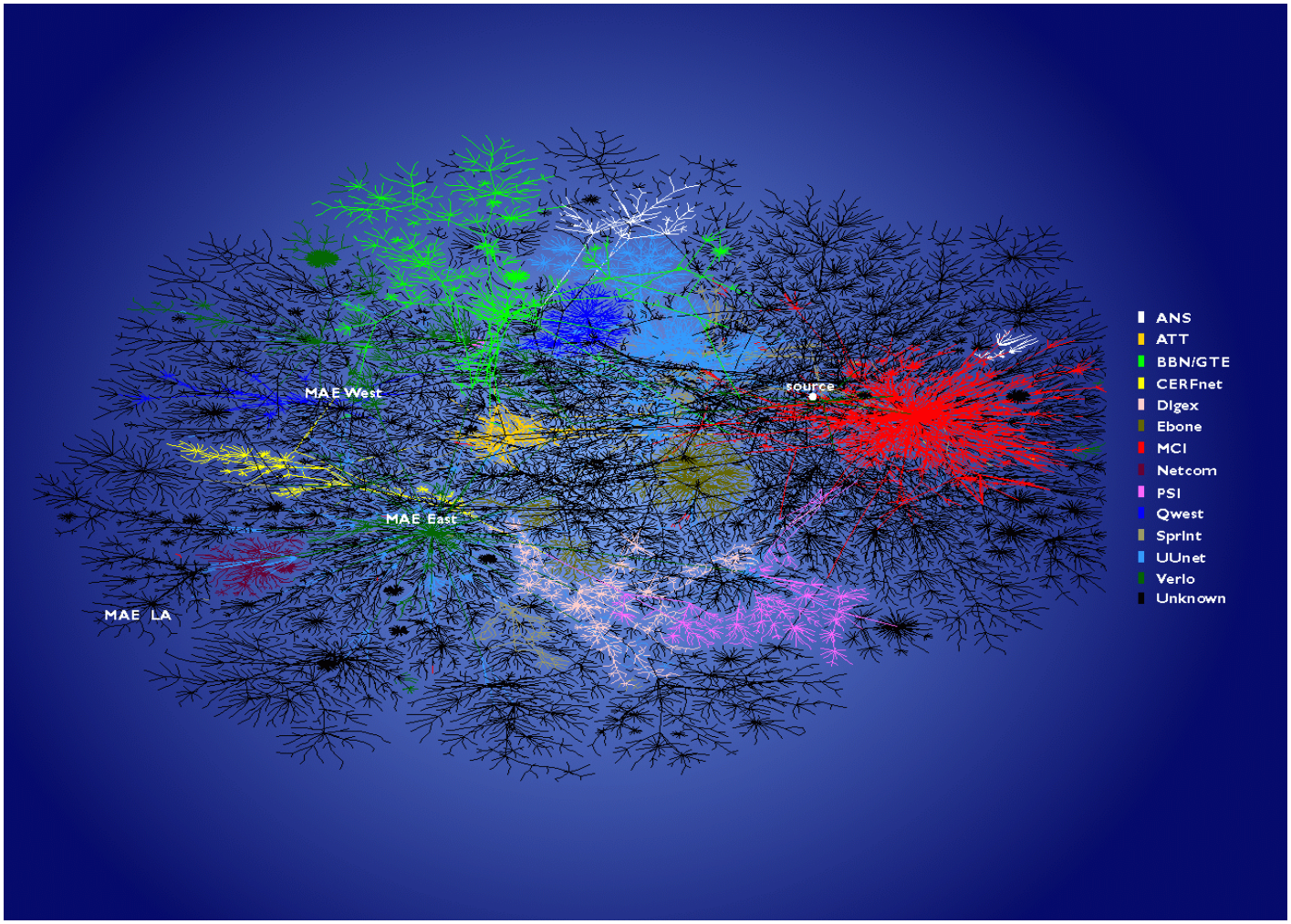


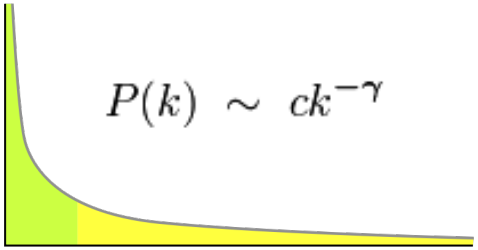
© C. Müller-Schloer 2015

- Artificial Life (AL) - Synthetic Life - Digital organisms
  - Creatures generated systematically by using computers
  - Not „wet“ biology
  - Have characteristics and capabilities of living systems.
- Not: „Life as it is.“ but „Life as it could be.“
- Strong ALife position: “Life is a process, which can be abstracted away from any particular medium.“
  - „Tierra is not simulating life, it **is** life“
- Weak ALife position: Mimick life processes in the computer
- Santa Fe Institute of Complexity: Complex adaptive systems (CAS)
  - Tom Ray, Chris Langton, Stuart Kauffman, Murray Gell-Mann, John Holland
- Applications??

© C. Müller-Schloer 2015

Small World: Internet





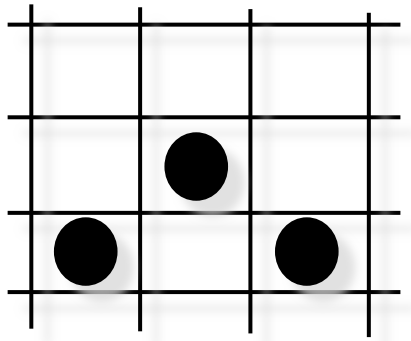
P(k): The fraction of nodes in the network having *k* connections to other nodes (for large values of *k*), where c is a normalization constant and $2 < \gamma < 3$

Skitter data depicting a macroscopic snapshot of Internet connectivity, with selected backbone **ISP**s (Internet Service Provider) colored separately
By **K. C. Claffy**:
kc@caida.org
http://www.caida.org/Papers/Nae/

© C. Müller-Schloer 2015

Cellular Automaton (1): Game of Life

- Field of Finite State Machines (FSM)
- Automaton changes its state depending on the states of its neighbours
- Example: Game of Life (John Conway)
  http://www.bitstorm.org/gameoflife

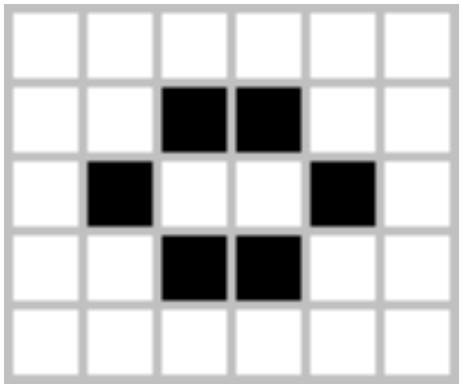


- Rules:

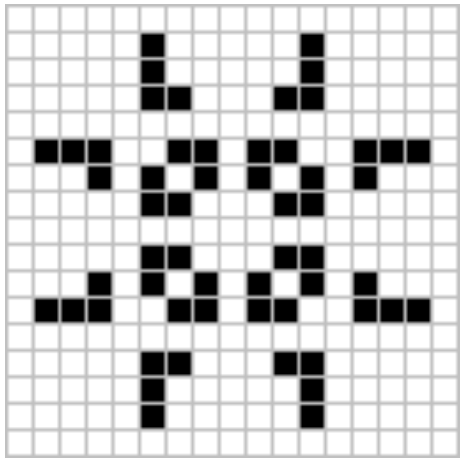
≤ 1 neighbour	dead
2 neighbours	const
3 neighbours	alive
≥ 4 neighbours	dead

© C. Müller-Schloer 2015

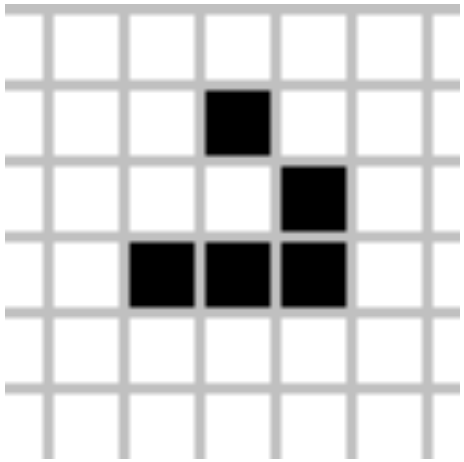
- Examples:



Static



Oscillating



Moving

Source: http://en.wikipedia.org/wiki/Conway%27s_Game_of_Life

- Generalisation based on: form, position, colour, communication, etc.
  => Used to simulate cell clusters

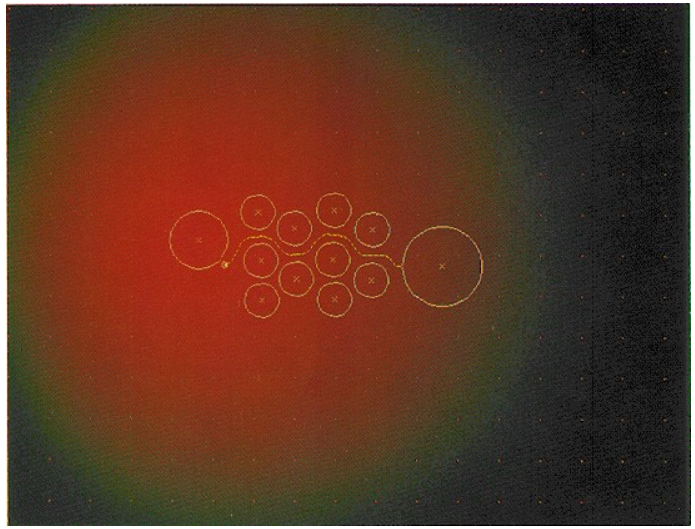
© C. Müller-Schloer 2015

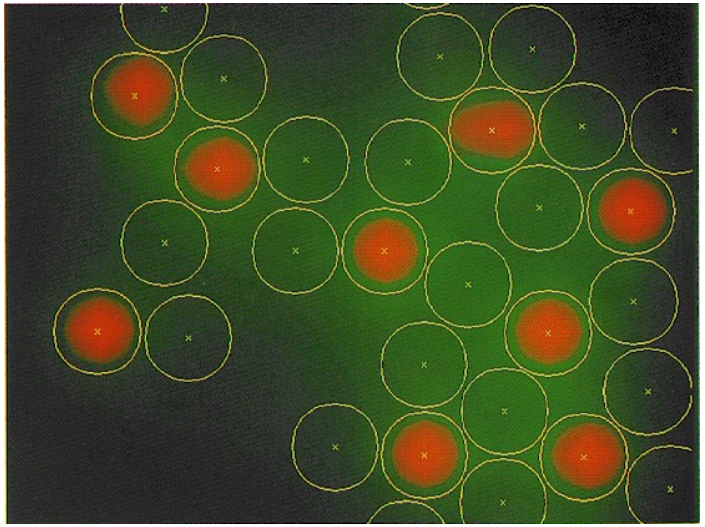
Cell:	Circle 	
Colour:	depends on current state	
Communication: (Cell – Cell)	collision, adhesion	mechanical
Environment:	Diffusion of chemicals Obstacles	chemical mechanical
Cell – Environment:	Absorption and emission Collision and adhesion between cell and obstacle	chemical mechanical

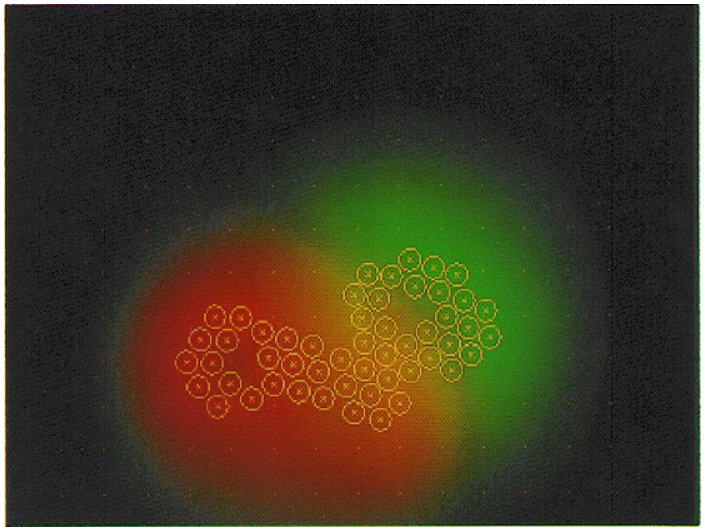
Fleischer, Barr

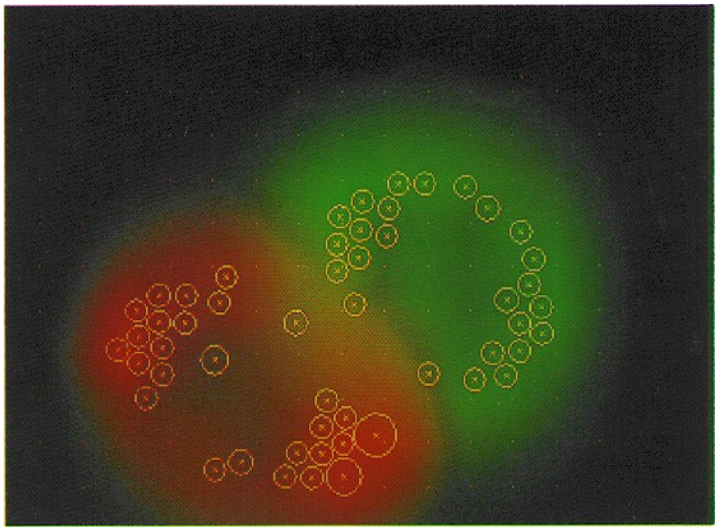
© C. Müller-Schloer 2015

Cellular Automaton (4): Examples of simulated cell clusters



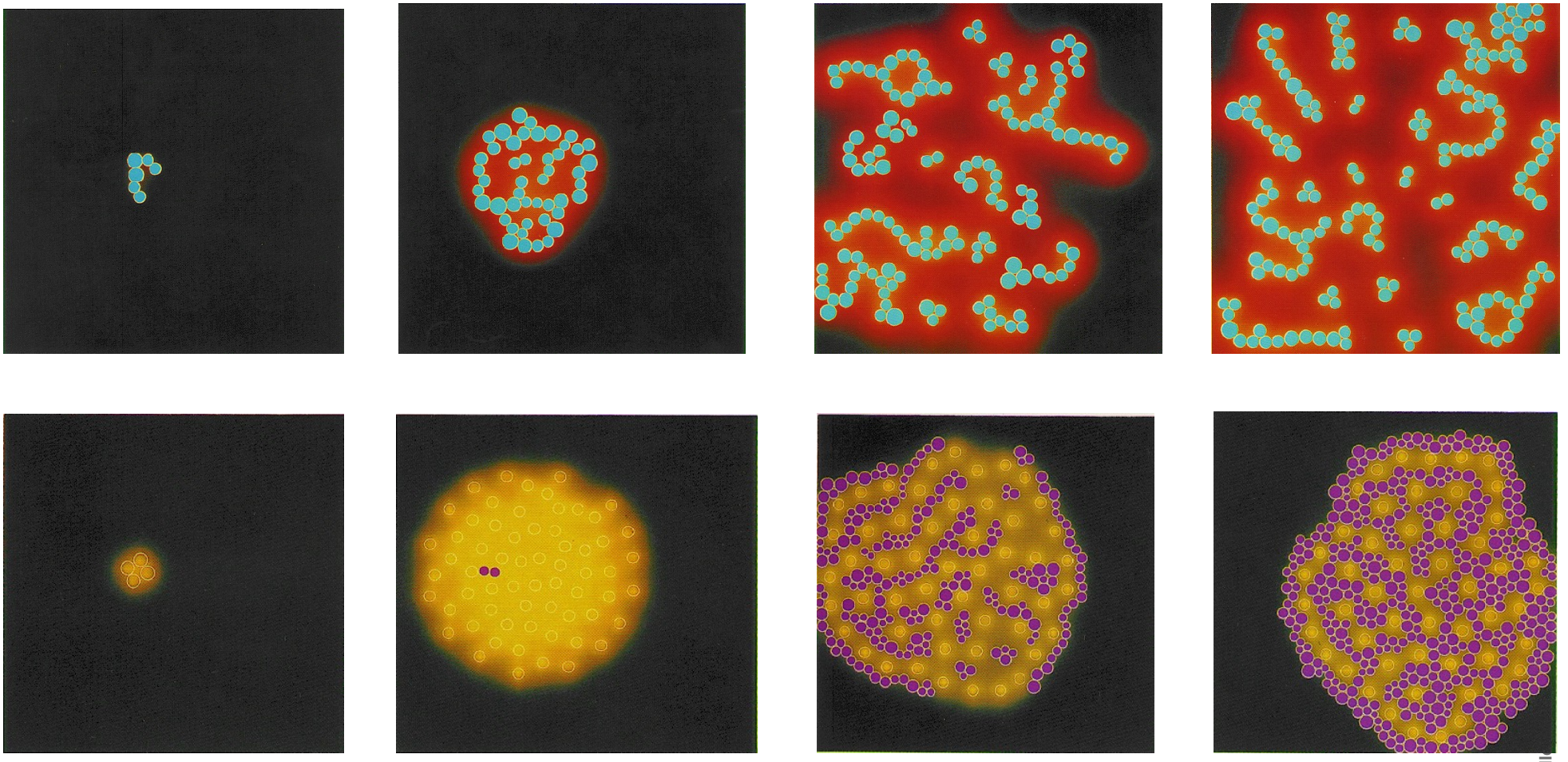






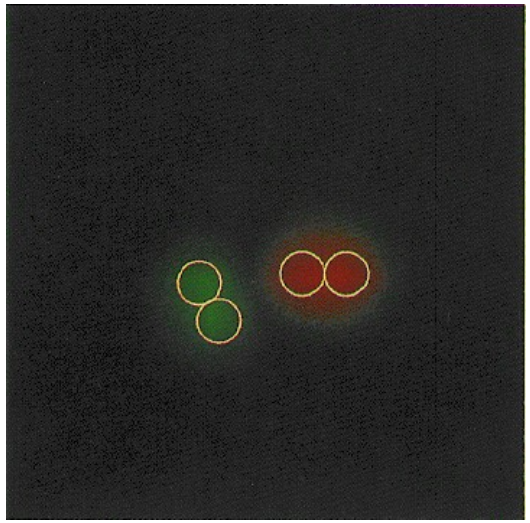
© C. Müller-Schloer 2015

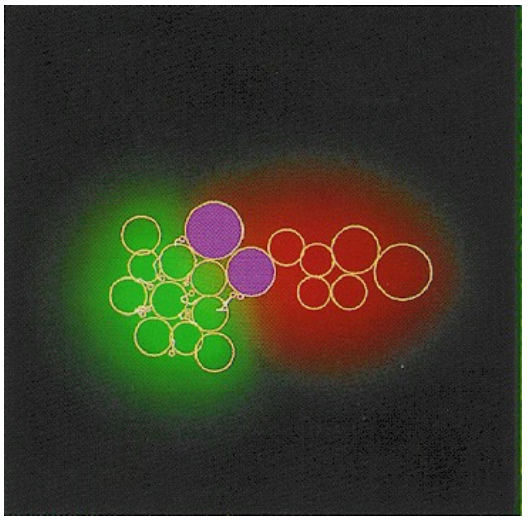
Cellular Automaton (5): Examples of simulated cell clusters

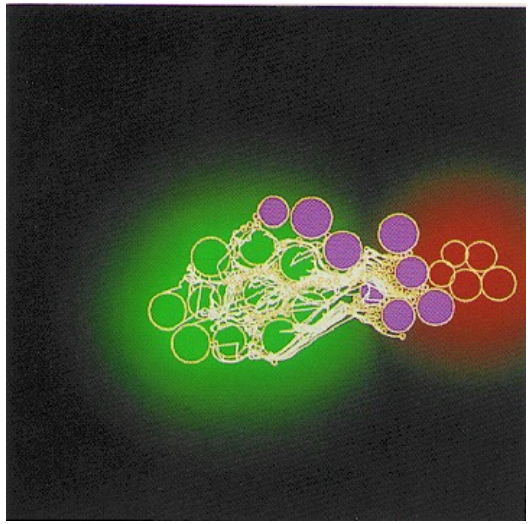


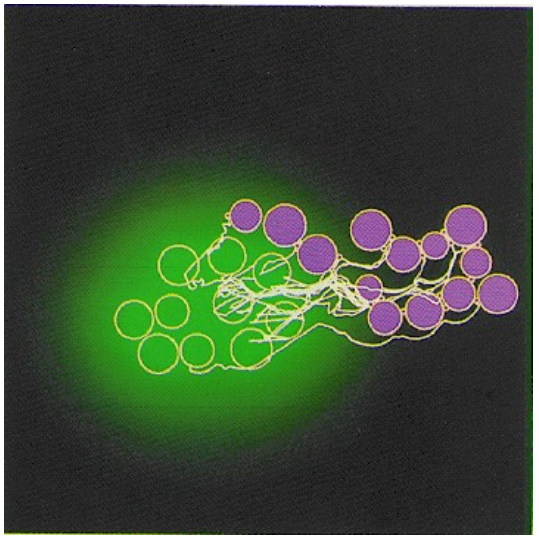
© C. Müll

Cellular Automaton (6): Examples of simulated cell clusters







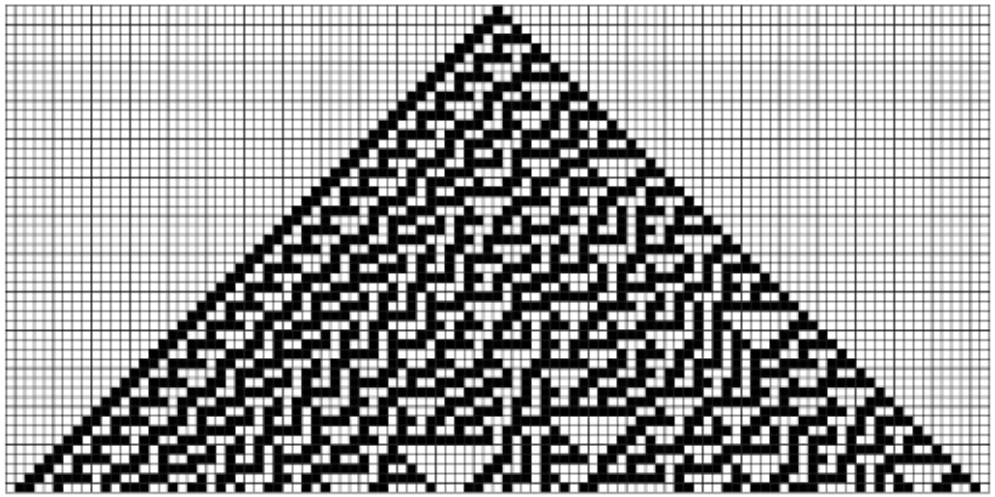


© C. Müller-Schloer 2015

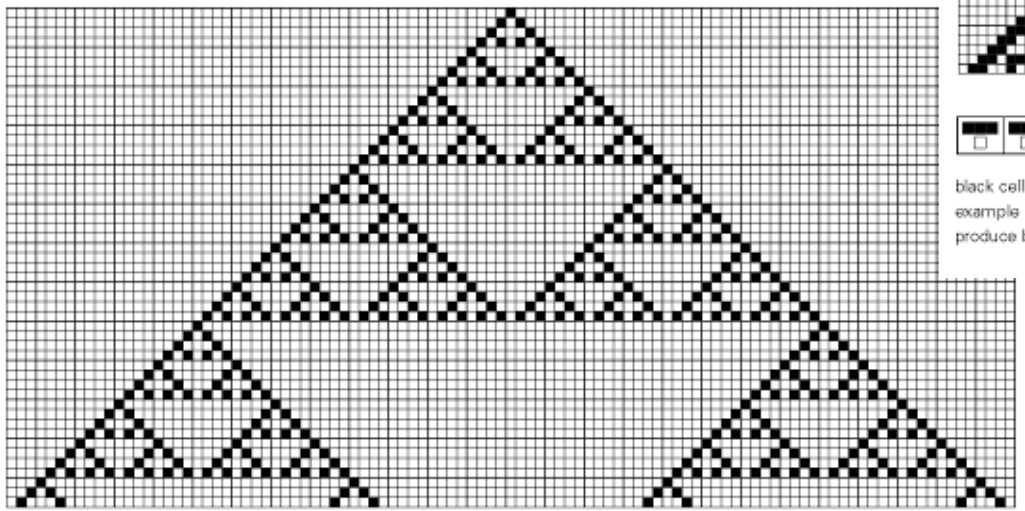
Example: Cellular automaton (Wolfram)

Stephen Wolfram:
A New Kind of Science

http://wolframscience.com



A cellular automaton with a simple rule that generates a pattern which seems in many respects random. The rule used is of the same type as in the previous examples, and the cellular automaton is again started from a single black cell. But now the pattern that is obtained is highly complex, and shows almost no overall regularity. This picture is our first example of the fundamental phenomenon that even with simple underlying rules and simple initial conditions, it is possible to produce behavior of great complexity. In the numbering scheme of Chapter 3, the cellular automaton shown here is rule 30.



A cellular automaton that produces an intricate nested pattern. The rule in this case is that a cell should be black whenever one or the other, but not both, of its neighbors were black on the step before. Even though the rule is very simple, the picture shows that the overall pattern obtained over the course of 50 steps starting from a single black cell is not so simple. The particular rule used here can be described by the formula $a_i' = Mod[a_{i-1} + a_{i+1}, 2]$. In the numbering scheme of Chapter 3, it is cellular automaton rule 90.

© C. Müller-Schloer 2015

- Lindenmayer-Systems (L-Systems) are parallel string-rewrite systems.
  - Compact way to describe iterative graphics using a turtle analogy.
  - Start with an axiom (a line segment) and one or more replacement rules.
  - Iterate several times, → complicated fractal curve
  - Axiom = F, replacement rule F → F+F--F+F

F

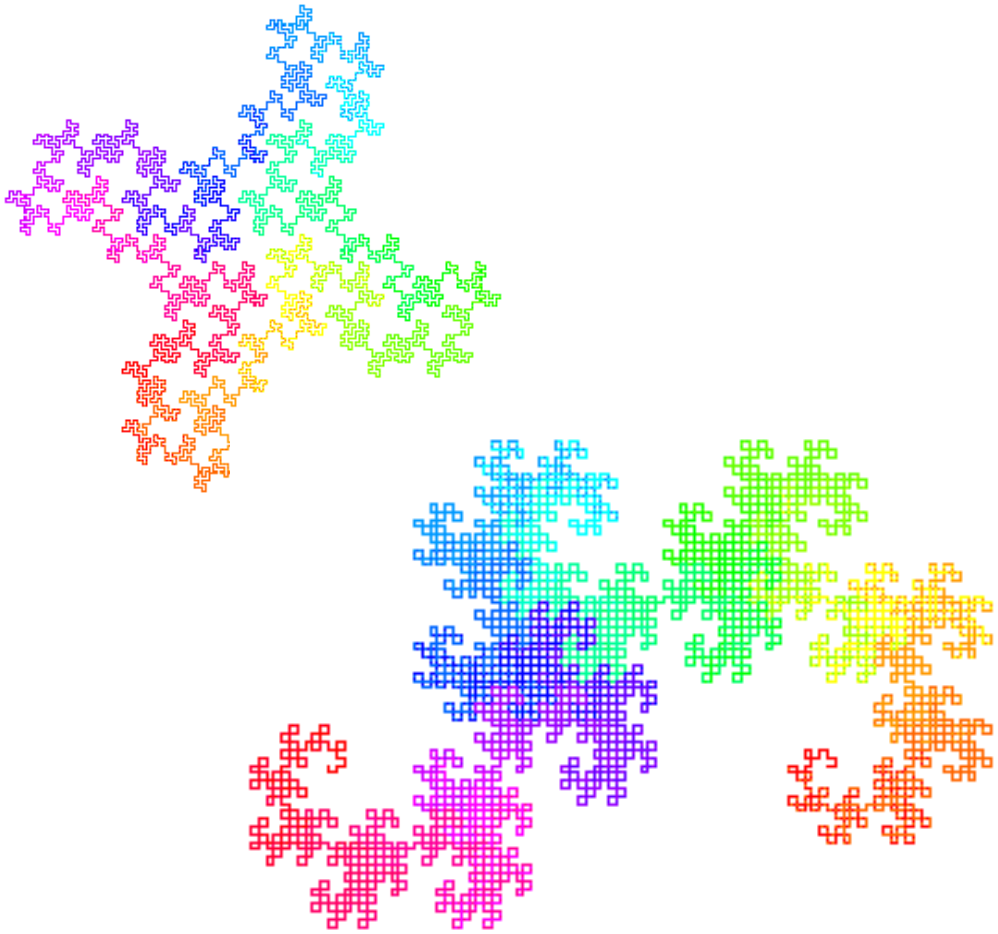
F+F--F+F

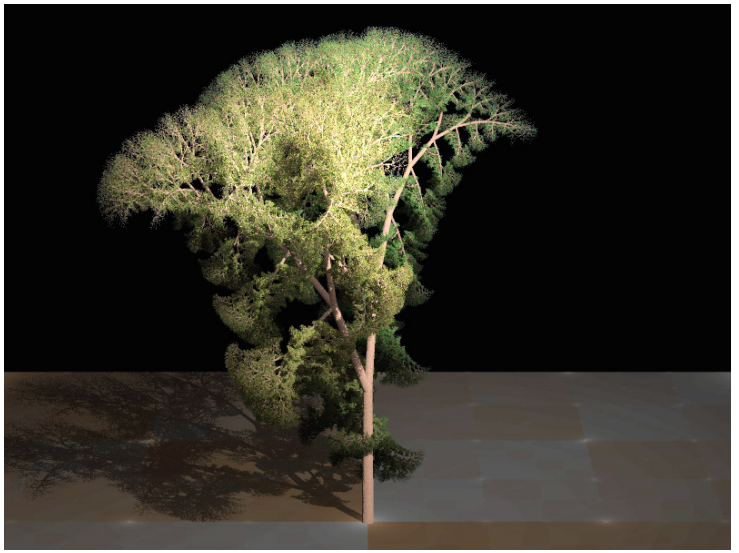
F+F--F+F+F+F--F+F--F+F--F+F+F+F--F+F

F+F--F+F+F+F--F+F-- F+F-- F+F+F+F-- F+F+F+F--F+F+F
+F--F+F--F+F--F+F+F+F--F+F--F+F--F+F+F+F--F+F--F+F--
F+F+F+F--F+F+F+F--F+F+F+F--F+F--F+F--F+F+F+F--F+F

© C. Müller-Schloer 2015

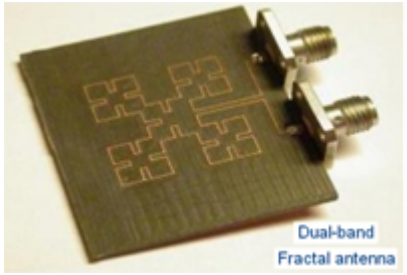
- Examples of more complex L-Systems

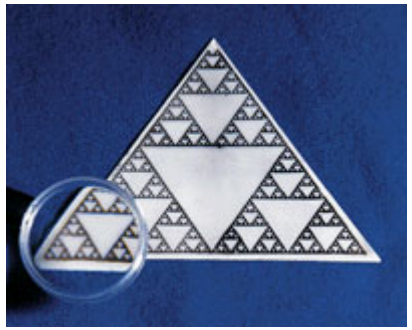


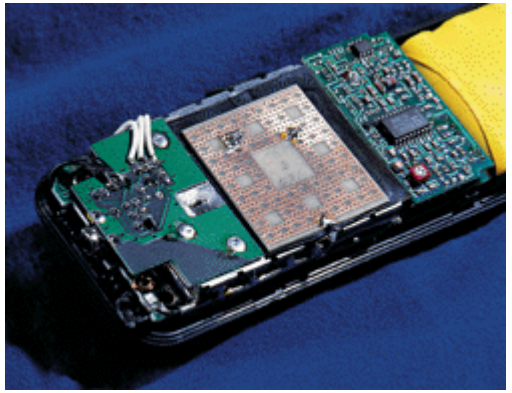


© C. Müller-Schloer 2015

- L-Systems in cell phones
  - Fractal Antennas allow for multiple frequency ranges
  - Quad-band phones: 850, 900, 1800, 1900 MHz







© C. Müller-Schloer 2015

Preconditions

--

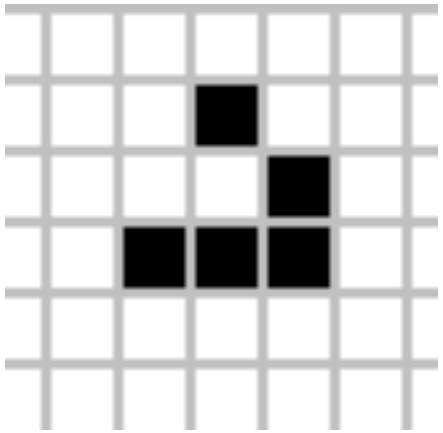
Objectives

Example-based introduction to Organic Computing

Content

- Examples
- Common characteristics
- Technical usage: Organic Computing

© C. Müller-Schloer 2015

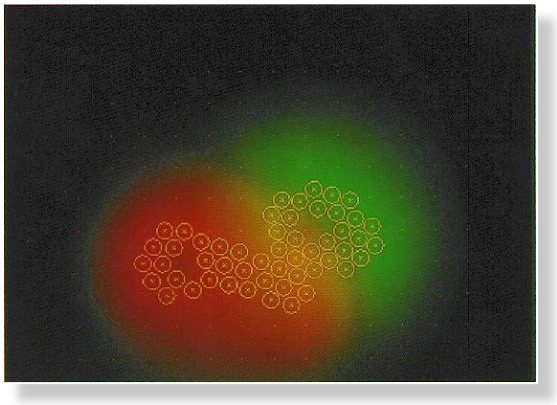


1. „Autonomous agents“ use local knowledge, behave locally.
2. Agents interact in large populations.
3. Learning by non-deterministic state exploration: evolution
4. Local behaviour leads to complex global behaviour: Emergence.
5. Global order evolves without control from „outside“: Self-organisation.

Local autonomy, interaction, evolution and non-determinism, emergence, self-organisation

What for?

© C. Müller-Schloer 2015



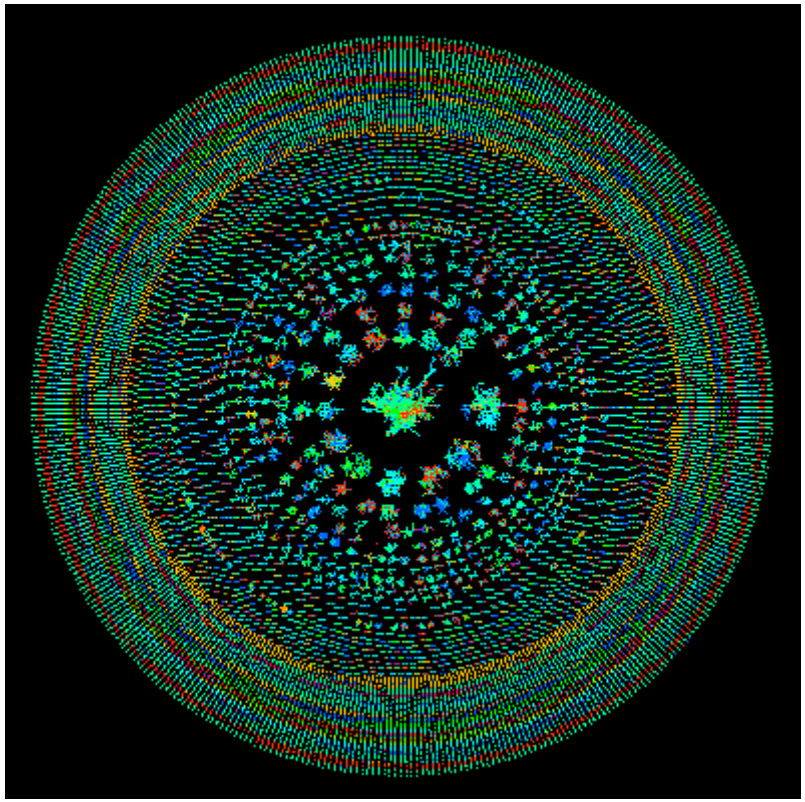
- Emergence
  - Emergent systems generate quasi-stable patterns.
  - Emergence leads to complexity-reduction (order).
  - System description: processes instead of structure
- Emergence occurs without control from „outside“.
- Emergence might be creative, but never targeted.
- Emergence can be part of technical systems, but it has to be controlled.

- Contradiction/conflict: bottom-up ←→ top-down

© C. Müller-Schloer 2015

- **Emergence**
  Local actions/local behavior of the members of a self-organizing system lead to observable, global patterns (structure or behavior).
  „The whole is more than the sum of its parts.“
- **Self-organization**
  Self-organization refers to a process in which the internal organization of a system, normally an open system, increases automatically without being guided or managed by an outside source.
- Self-organizing systems typically (though not always) display emergent properties.

- **Emergence = self-organized order**



Small World Effect: Supermarket
A set of 10 million receipts from a large DIY store were processed so as to link together items that often appear on the same receipt.
By **Graham Wills**: gwills@research.bell-labs.com
http://www.bell-labs.com/user/gwills/NICHEguide/bs.html

© C. Müller-Schloer 2015

Preconditions

--

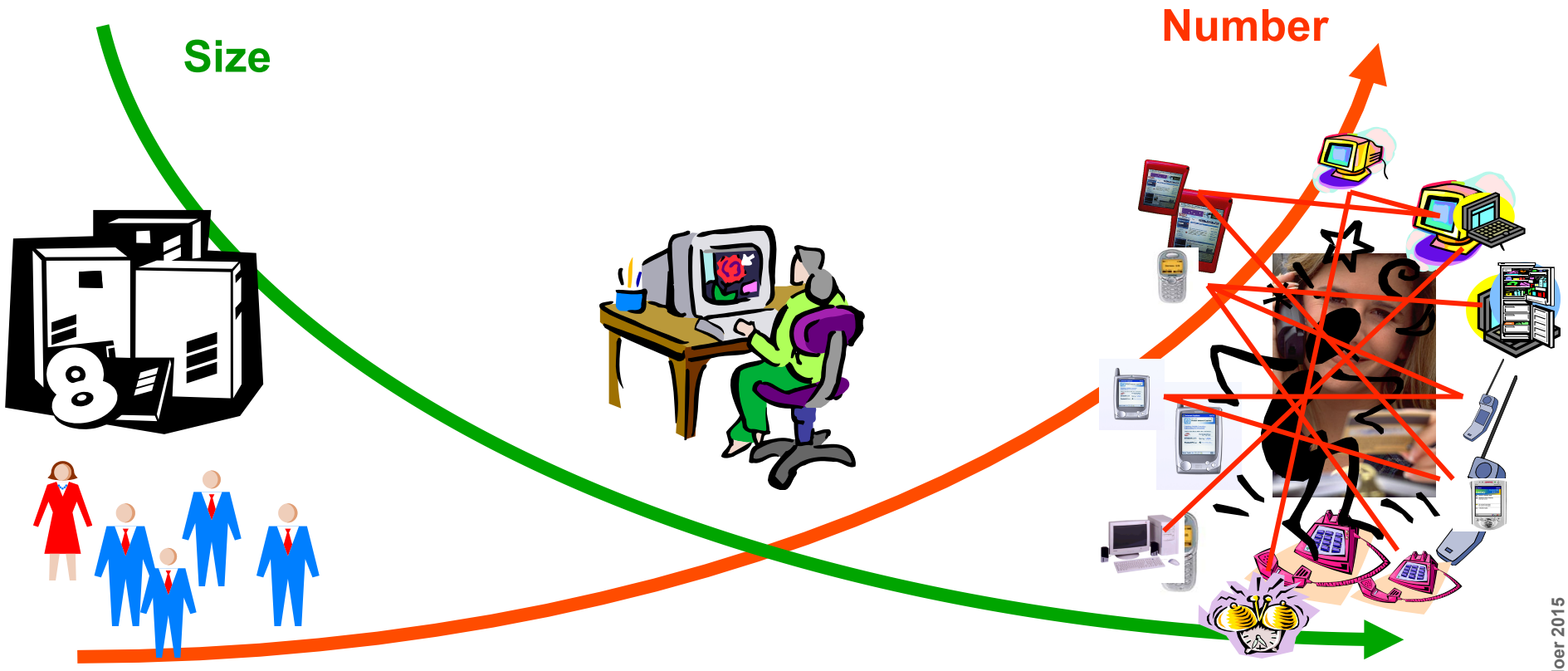
Objectives

Example-based introduction to Organic Computing

Content

- Examples
- Common characteristics
- Technical usage: Organic Computing

© C. Müller-Schloer 2015

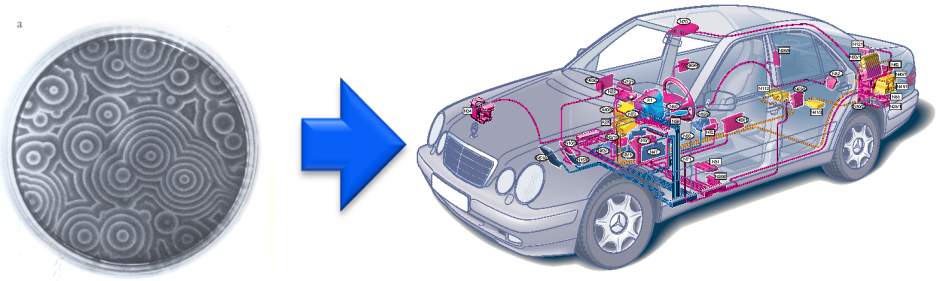


© C. Müller-Schloer 2015

Local autonomy, interaction, evolution and non-determinism, emergence, self-organisation

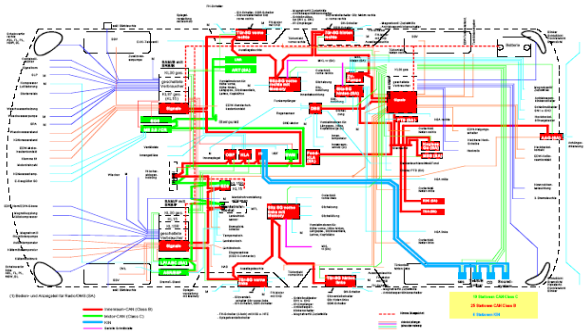
What for?

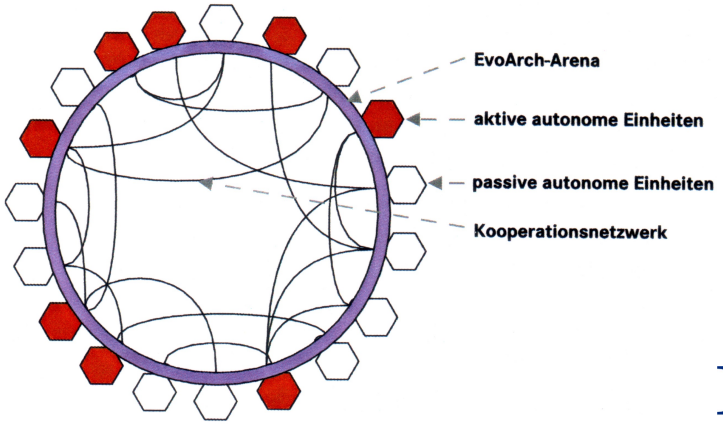
- The problem: Complexity of technical systems
- Technical systems
  - Large populations ✔
  - Autonomy ✔
  - Interaction ✔
  - Locality ✔
  - Adaptivity and learning ✘
  - Evolution ✘
  - Self-organization ✘
  - Emergence ?
- → Apply natural principles to technical systems!



© C. Müller-Schloer 2015

- Self-organisation in technical systems with a restricted, defined configuration space: EvoArch (Hofmann, 2001)
  - Autonomous entities
  - Decentralized operation
  - Ad-hoc confederations
  - Cooperation based on contracts
  - Taxonomy defines relations
  - Autonomous entities possess motivation





© C. Müller-Schloer 2015

DAIMLERCHRYSLER

ISE SRA

- Motivation
  Due to the increasing complexity of technical systems, standard top-down methods will not be sufficient to solve upcoming problems (and therefore develop future IT systems)!
- Solution: Organic Computing (OC)
  - New research area focusing on self-organized systems
  - Autonomous entities act without strict central control.
  - Decisions are mainly based on local knowledge (leads to global behaviour).
  - Complexity => not all situations can be foreseen (development process).
    - Ability to learn new actions and strategies for previously unknown situations
    - OC systems must be adaptive and equipped with learning capabilities.
  - OC systems possess self-x properties.

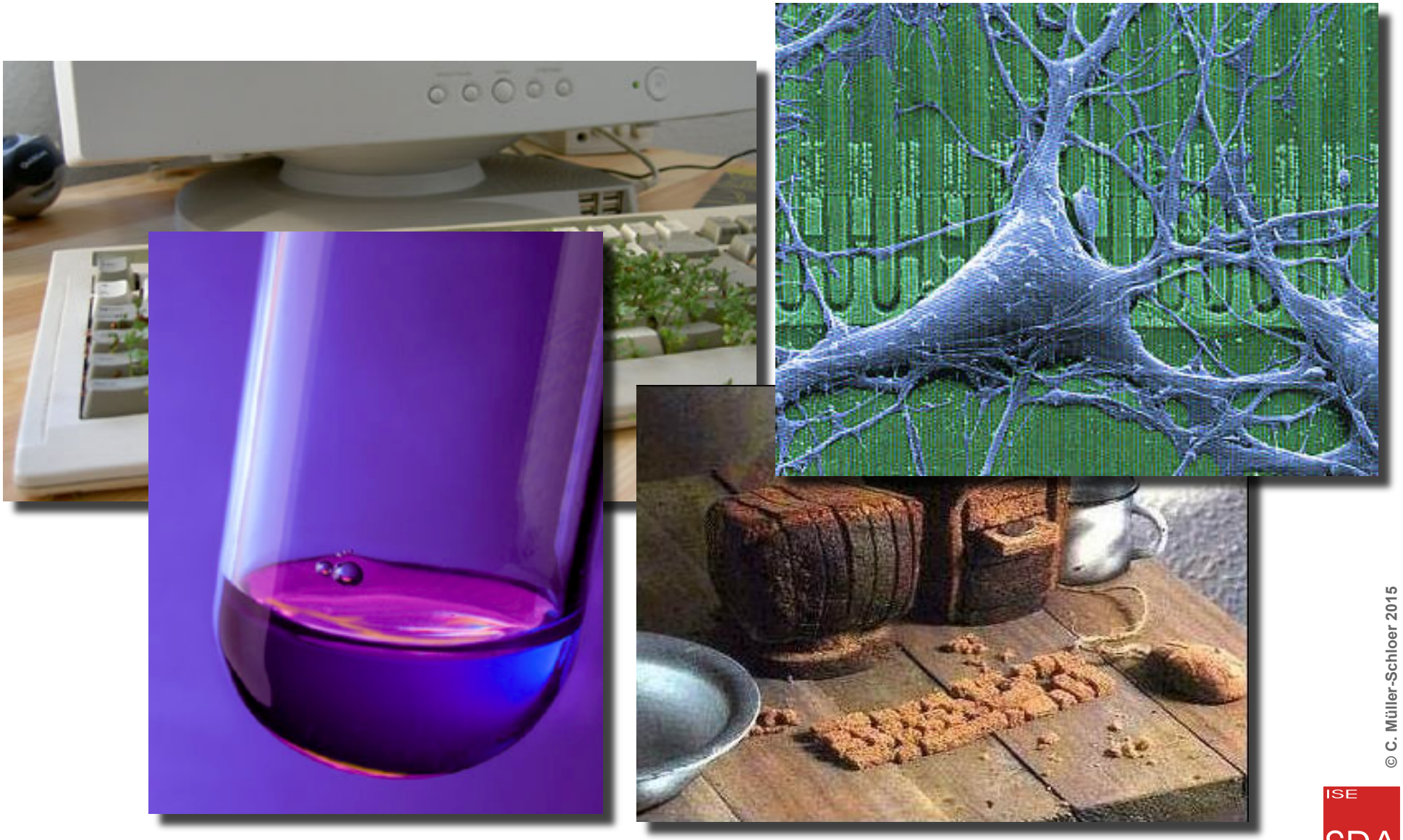
© C. Müller-Schloer 2015

- Organic Computing systems possess life-like characteristics.
- Self-organisation allows for an adaptive and context-sensitive behaviour:
  - Self-configuring
  - Self-optimizing
  - Self-healing
  - Self-stabilizing
  - Self-protecting
  - Self-describing
  - …
- Self-X

Basis for OC systems is still silicon!

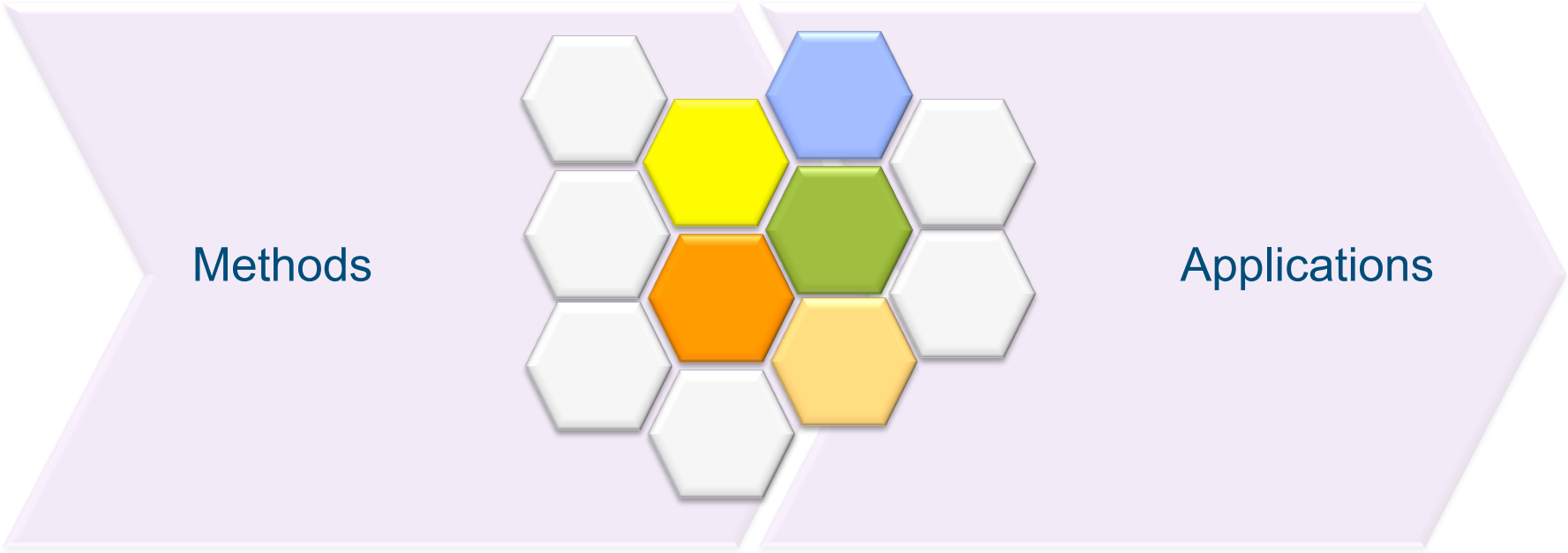
© C. Müller-Schloer 2015

This is not Organic Computing!

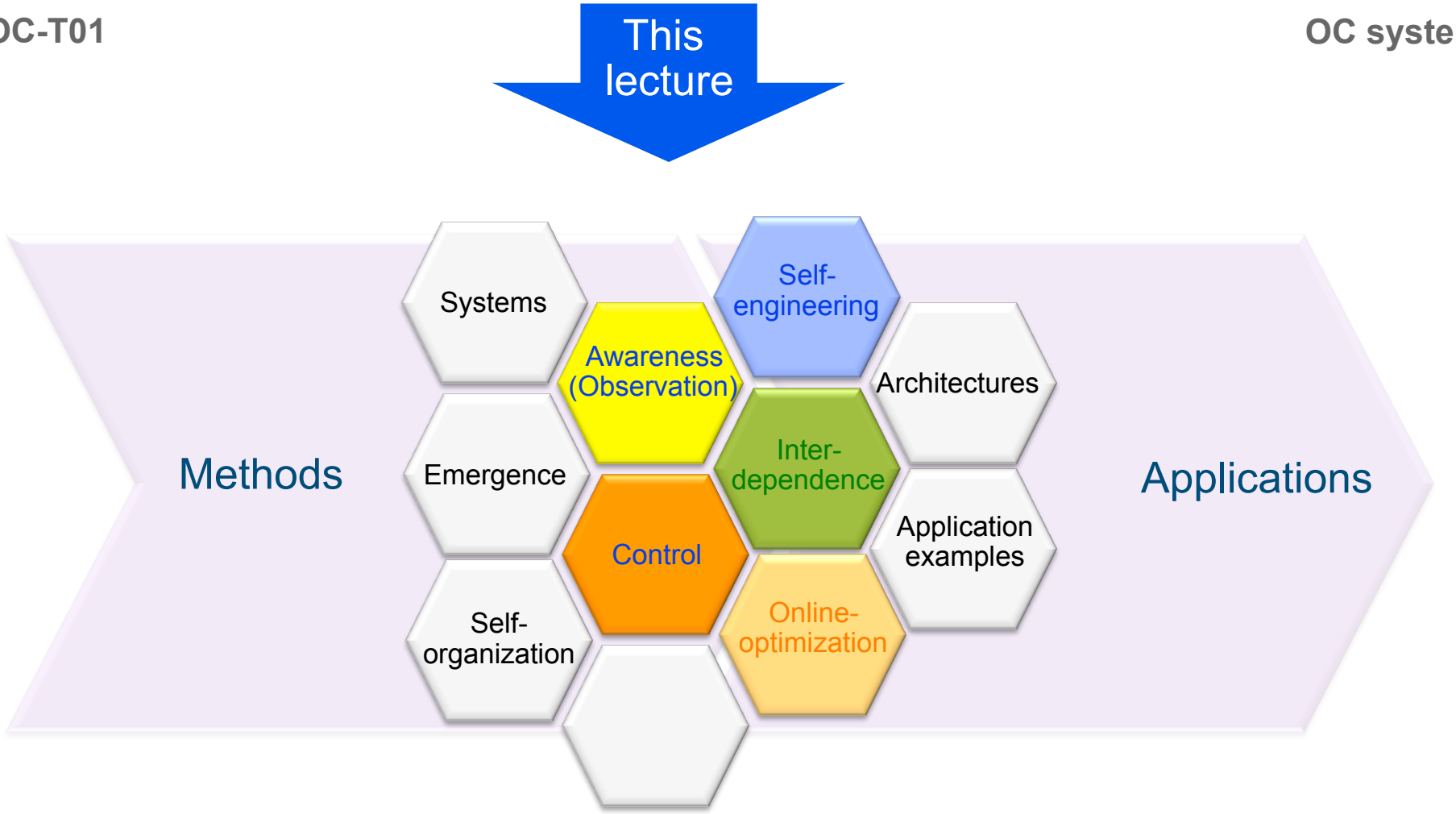


© C. Müller-Schloer 2015

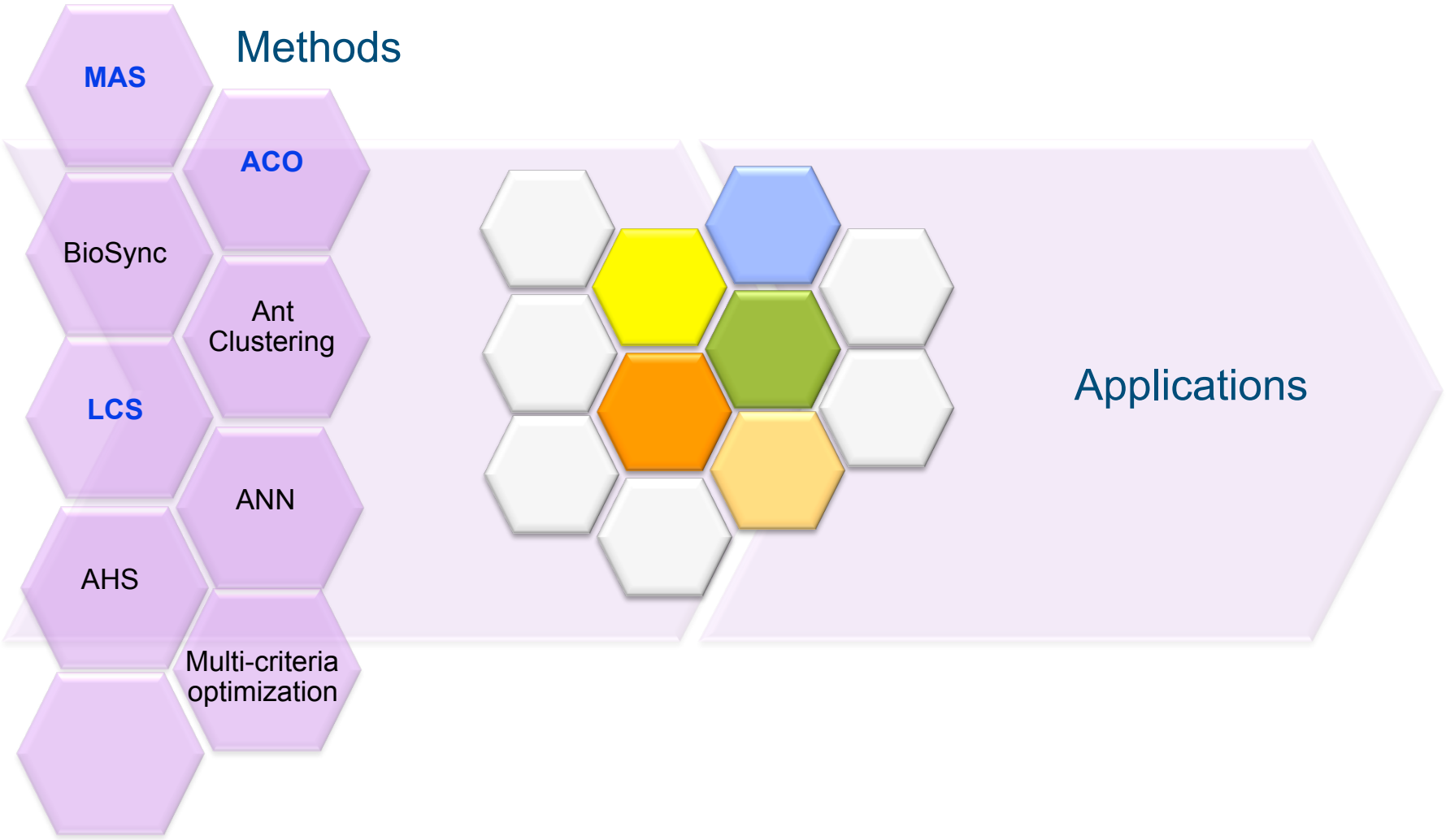
OC kernel competencies



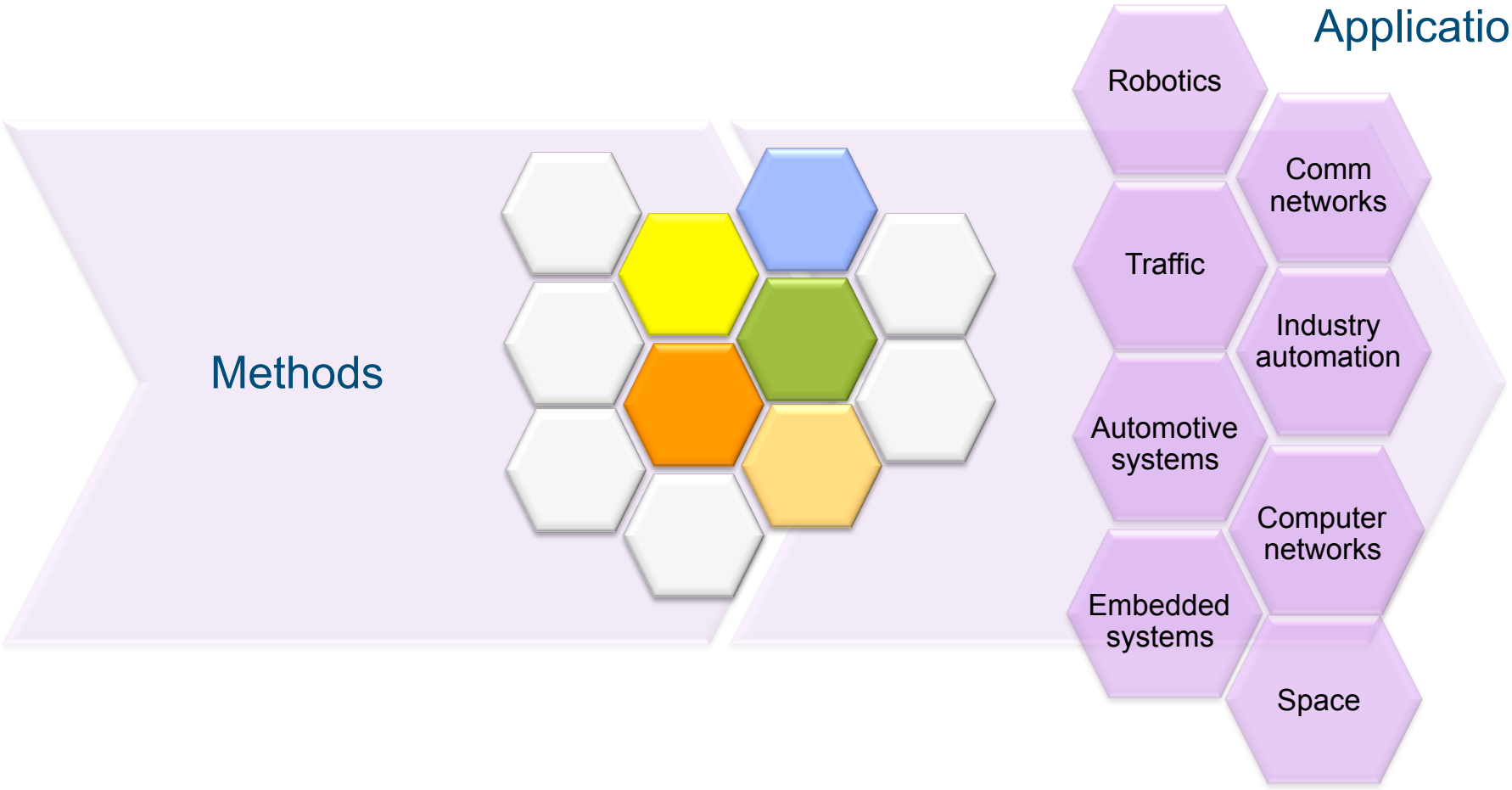
© C. Müller-Schloer 2015



© C. Müller-Schloer 2015



© C. Müller-Schloer 2015



© C. Müller-Schloer 2015

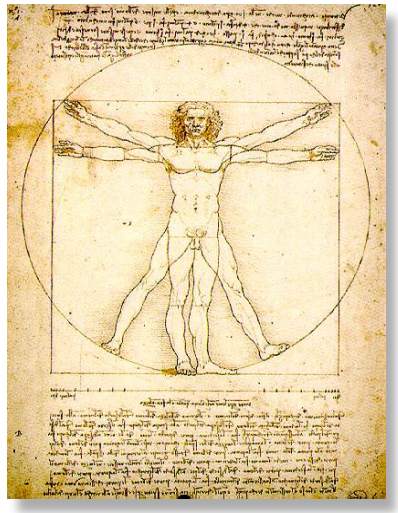
The increasing complexity of technical systems is not controllable anymore – if we keep on using purely hierarchical, linear, top-down and exact causal-logic approaches of thinking and developing.

It is not the question,
whether adaptive and self-organizing systems will emerge,
but ***how*** we will cope with them.

© C. Müller-Schloer 2015

Where do we want to go?

- Do anything technically feasible?
- Focus on desirable goals!
- Humans serve computers? (German: „Bediener“)
- Computers serve humans!



NATURAM OBSERVANTES
VIVERE DISCIMUS

Organic Computing

© C. Müller-Schloer 2015