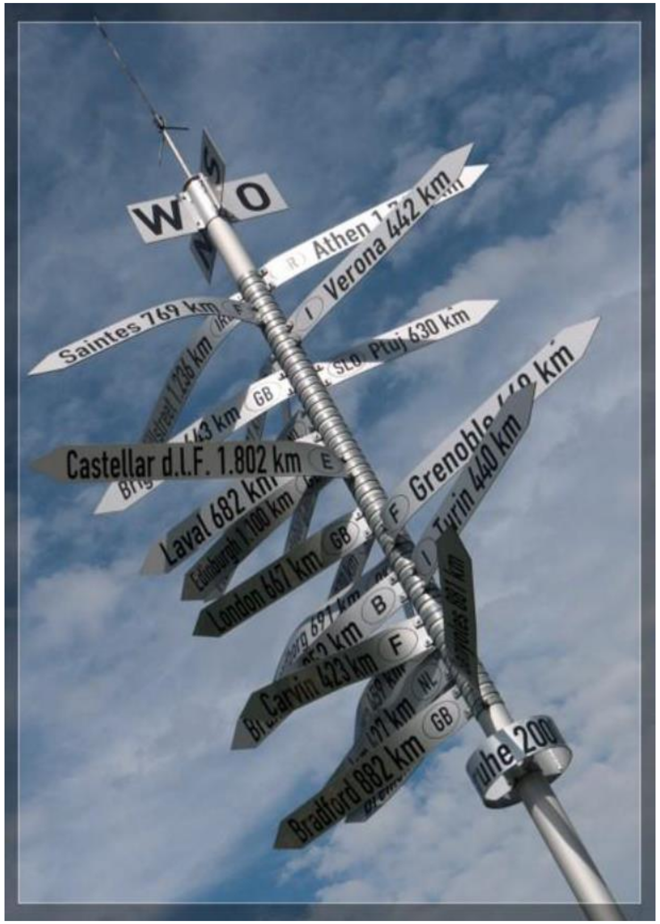
Agenda

- A first example: water temples in Bali
- A second example: ants
- Emergence
- Term definition
- Quantification of emergence
- A refined approach to emergence quantification
- Conclusion and further readings



Emergence in Organic Computing / intelligent systems

- Why do we have to consider emergence in OC systems?
- Do we want do explicitly design emergent effects?
  → Hardly possible…
- Emergence is not something we want to design, but something that will appear automatically!
  - Emergence is the result of interactions between a set of self-organised entities.
  - OC systems consist of a set of self-organised interacting entities.
    → We have to be aware of emergence: positive and negative!
  - In technical systems: How to be aware of something?
    → We need to measure it!

Structural Emergence

- Assumption: a number of similar individuals interacting
- (Structural) emergence shows as:
  - patterns in time and/or space
  - patterns (order) at the system level.
  - patterns have properties not existent in the individuals

- How can we measure emergence?
  - Patterns $\Leftrightarrow$ order
  - Entropy is a measure of order! $\Rightarrow$ How is emergence related to entropy?
  - Note: Order per se says nothing about self-organisation.

Quantification of emergence

Goal: Assign a high emergence value to a system, which is perceived as emergent!

Approach:

- Basis: non-formal definitions
- Emergence is always associated with patterns (symmetry breaks).
- This corresponds to structural emergence.
- Patterns represent order.
- Order can be measured in terms of entropy (inversely proportional).
- Therefore, we must 1) define entropy and 2) relate it to emergence!

What is order?

Where is more order? Left or right?

- Right: higher entropy
- Left: more structure

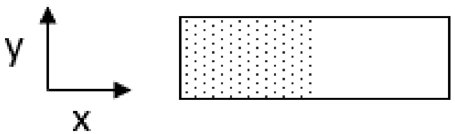
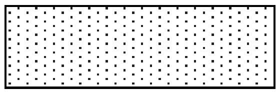
ORGANIC COMPUTING

Order is subjective!

- The perception of order depends on the view of the observer.
- The purpose and the sensory equipment of the observer determine the view.
- A system can be rated as orderly or disorderly dependent on the utility.

Views and order

- “Order” or “disorder” depends on
  - the purpose and
  - the view (aspect).
- A view is determined by the selection of certain attributes (or a group of attributes) of an object.
- Example:

View	x position	colour
	higher order	same order
	lower order	same order

- The view is influenced by the pre-processing of sensory data.

- Entropy is a thermodynamic state variable.
  - High entropy ⇔ high probability.
  - Clausius: Entropy of a closed system never decreases.
- Entropy: measure of (dis)order (high entropy = low order).

$$\boxed{S = k_B \cdot \ln(\Omega)}$$

- Statistical definition of entropy (S):
  - $k_B = 1{,}38 * 10^{23} \frac{J}{K}$ (the *Boltzmann* constant, “average kinetic energy of an ideal gas particle at a temperature of 1 Kelvin”)
  - $\Omega =$ probability of the current macroscopic state (= the current number of possible states of the particles in the system / total number of possible states)

Example: Diffusion

- Right: higher Ω, higher entropy
- Left: lower Ω, lower entropy

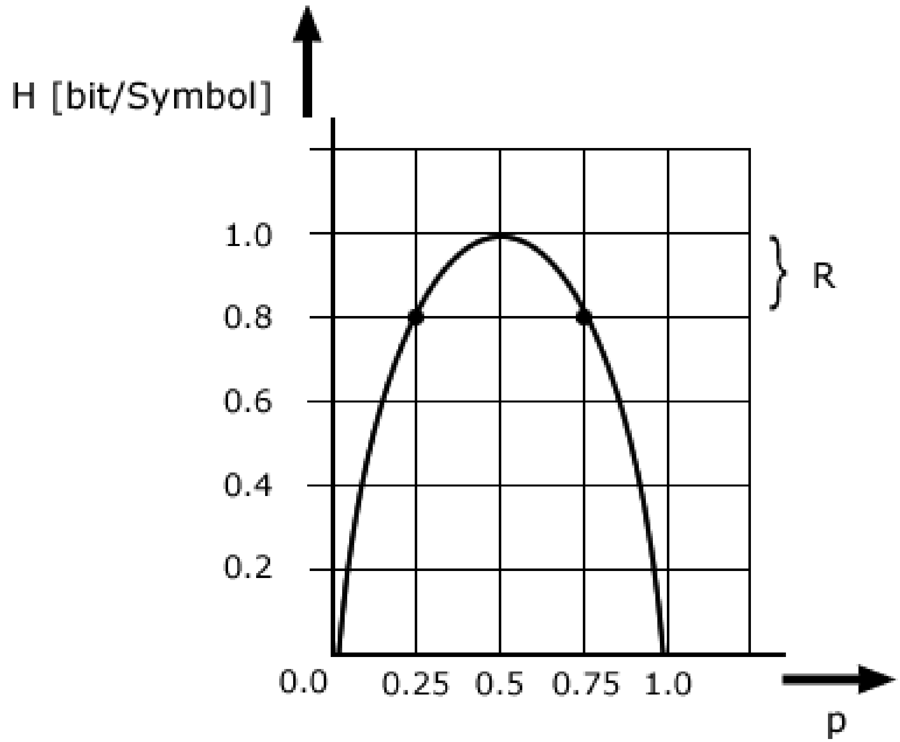
Definition from information theory (Shannon):

- Entropy is a measure of information.
- Message source M, alphabet Z:

$$\boxed{Entropy\, H(M) = -K \cdot \sum_{j=1}^{|Z|} p_j \cdot ld\, p_j}$$

- p_j = probability for occurrence of symbol $z_j \in Z$ in message source M
- Entropy H is a measure for random information in a system (or a message source M).
- K is a constant (can be neglected).
- High information content ⇔ low predictability
- Low information content ⇔ high predictability

- Example:
  - Stream of 2 symbols (0 and 1)
  - with probabilities p and (1 - p)
  - H = - p ld p - (1 - p) ld (1 - p)



Shannon's entropy (3)

- $H = H_{max}$ is desirable, if a channel must transport the maximal "newness" value per (physical) step.

- In case of $H < H_{max}$
  - The channel transports useless information (redundancy $R = H_{max} - H$).
  - Known information burdens the channel but does not increase the knowledge of the receiver.

- A Shannon channel is "good", if it transports the maximum amount of information:

  $\rightarrow R = 0$!

Approach: Use the statistical definition of entropy!

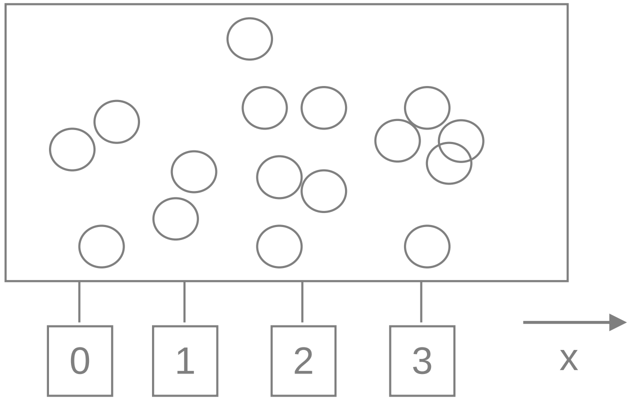
1. Select a (discrete, enumerable) attribute A of the system with possible values a_j.
2. Observe all system elements e_i and their respective value of A (one a_i for each e_i).
3. Transform into a probability distribution over the attribute values a_j (e.g. relative frequency).
4. Determine $H_A = -\sum_j p_j \operatorname{ld} p_j$

System entropy

- Each attribute X has some entropy H_X.
- System entropy: $$H_S = \sum_X H_X$$
- Characterisation of a system by:
  a) System entropy (however: low expressiveness)
  b) Vector of attribute entropies (fingerprint): $(H_A, H_B, H_C \ldots)$

Example for the quantification of emergence

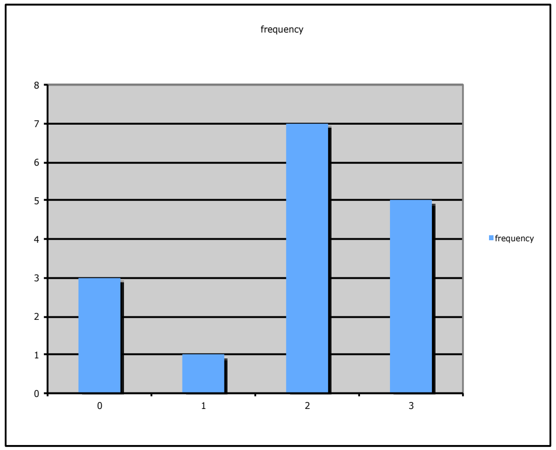
1. Chosen attribute: discrete value of x coordinate { 0, 1, 2, 3 }



2. and 3. Count and calculate relative frequency.

N = 16

Position:	0	1	2	3
Frequency:	3	2	6	5
p	3/16	2/16	6/16	5/16



4. Calculate entropy

$$H_x = -\left(\frac{3}{16} ld \frac{3}{16} + \frac{2}{16} ld \frac{2}{16} + \frac{6}{16} ld \frac{6}{16} + \frac{5}{16} ld \frac{5}{16}\right)$$
$$= 1{,}72 bit / element$$

Emergence definition (first try)

- Definition: Emergence (first try)
  - Emergence is a decrease of entropy over time (from a start state to an end state):

    $M = \Delta H = H_{Start} - H_{end}$

    $\Rightarrow$ Emergence ($\Delta H > 0$) if order increases $H_{End} < H_{Start}$
  - The process that leads to this must be self-organised (not e.g. human-induced).

- Problem: The observation of emergent phenomena frequently involves a change of abstraction level.

Emergence definition

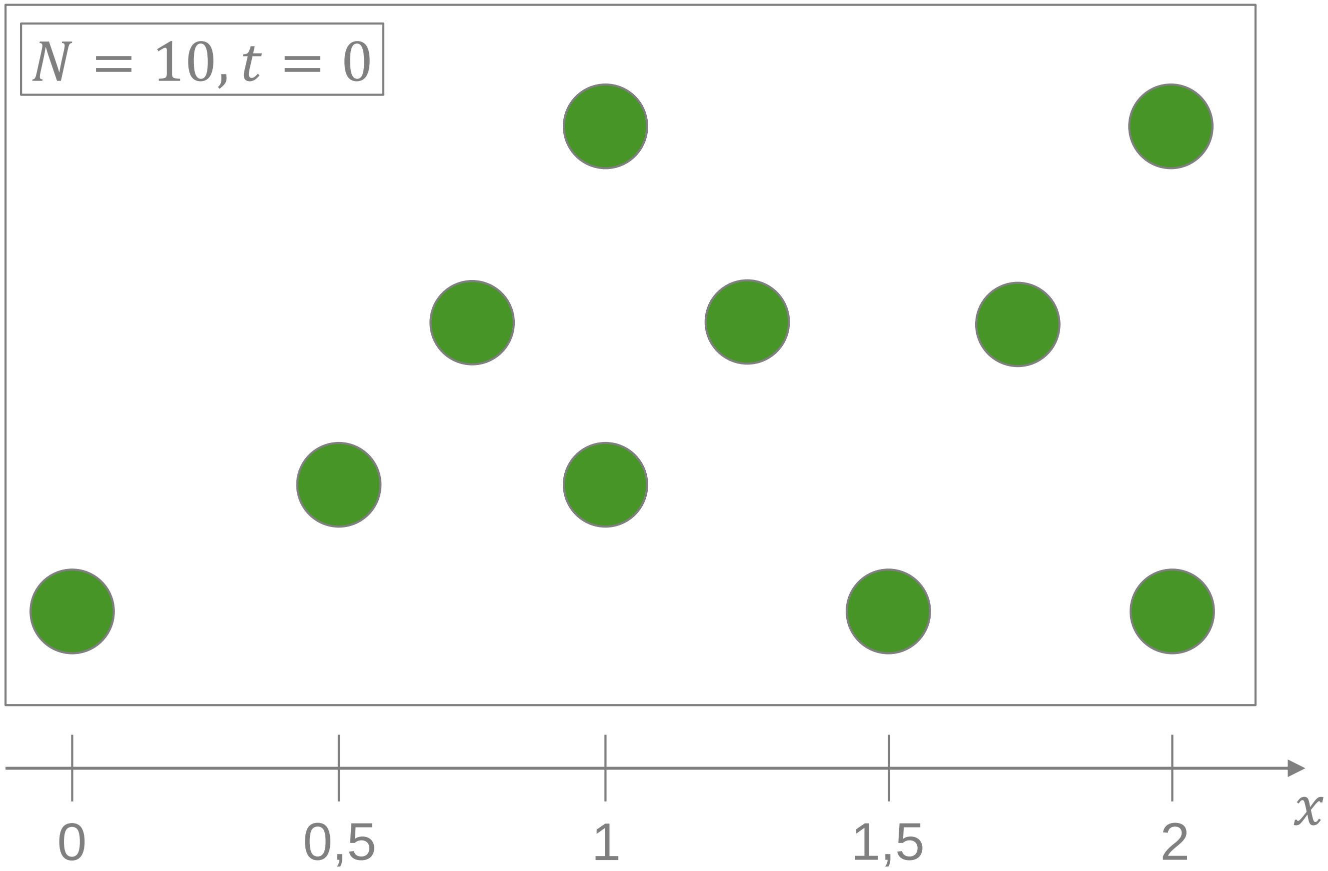
- Change of view to a higher abstraction level
  ⇒ positive ΔH (thus "higher order") but not due to an emergent process!
- Thus: $\Delta H = \Delta H_{view} + \Delta H_{emergence}$
- Definition: Emergence
  - Emergence is a decrease of entropy over time (from a start state to an end state):
    $M = \Delta H_{emergence} = \Delta H - \Delta H_{view} = H_{start} - H_{end} - \Delta H_{view}$
  - The process that causes this must be self-organised (not e.g. human-induced).

Example: Abstraction level change

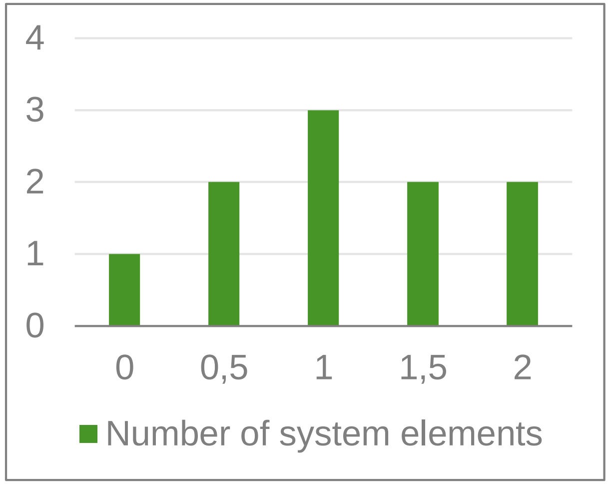
- Observation 1 of x coordinate: 32-bit floating point
- Observation 2 of x coordinate: quantization to 256 values (8 bit integer)
- Quantization results in entropy difference:
  - $\Delta H = 24 \frac{bit}{element}$
  - $\Delta H = \Delta H_{view}$
  - $Emergence\ M = \Delta H_{emergence} = 0$

Quantification of abstraction change

Consider a system S at time $t = 0$



attribute value	frequency
0	1
0,5	2
1	3
1,5	2
2	2

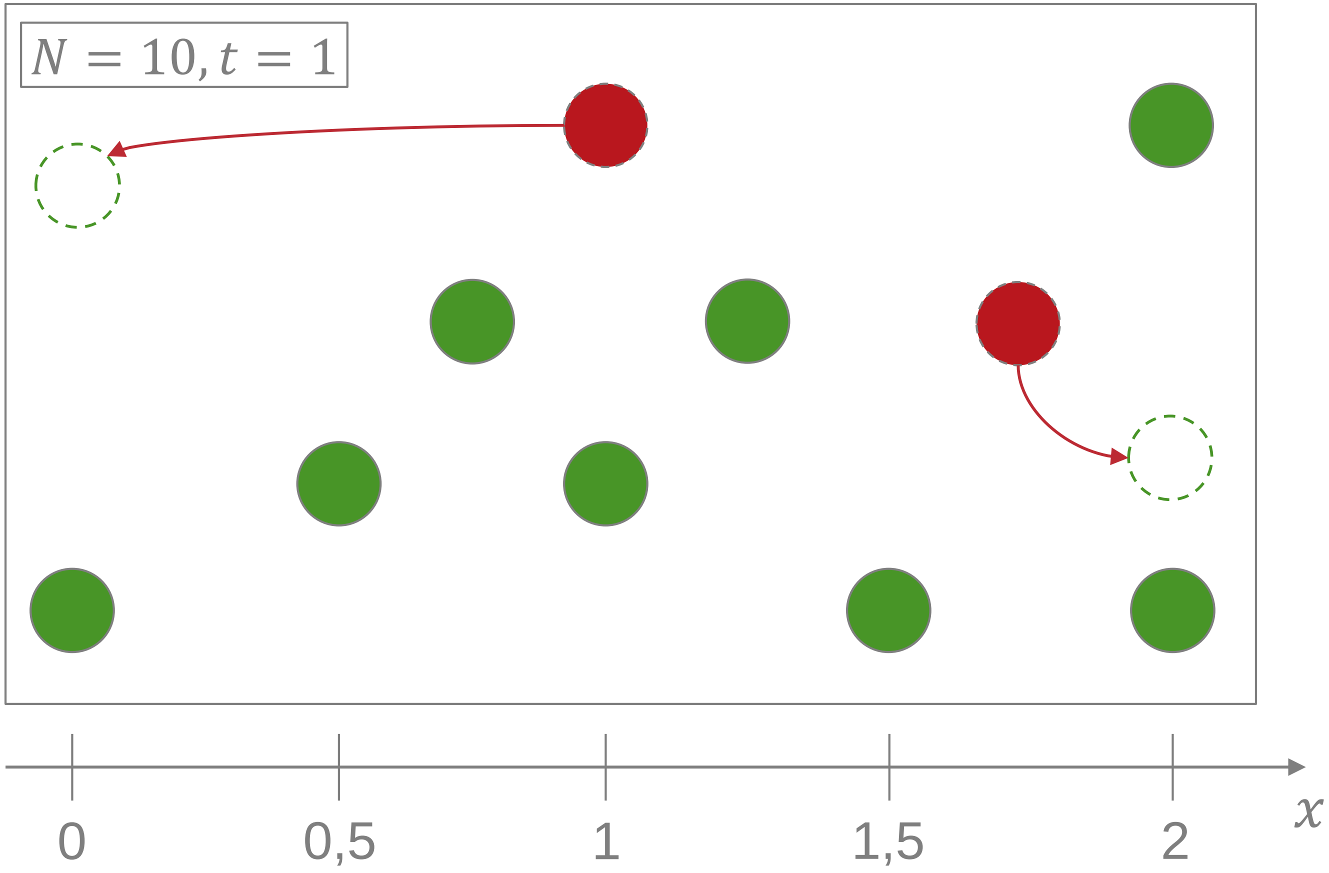


$$H_x^0 = -(\frac{1}{10} * ld\frac{1}{10} + 3 * \left(\frac{2}{10} * ld\frac{2}{10}\right) + \frac{3}{10} * ld\frac{3}{10})$$

$$H_x^0 = 2{,}24643934467$$

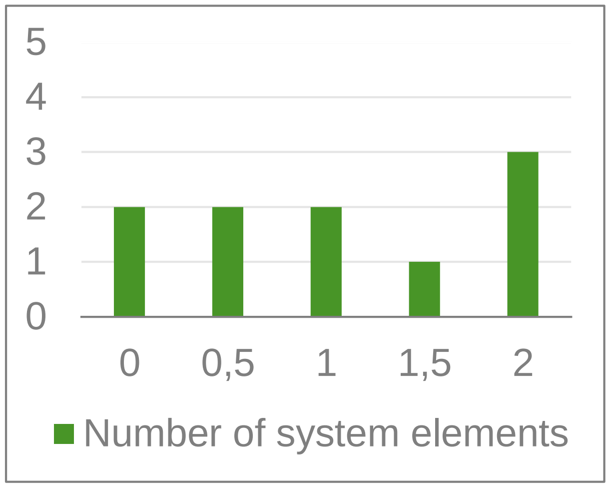
Quantification of abstraction change (2)

Something happened from $t = 0$ to $t = 1$



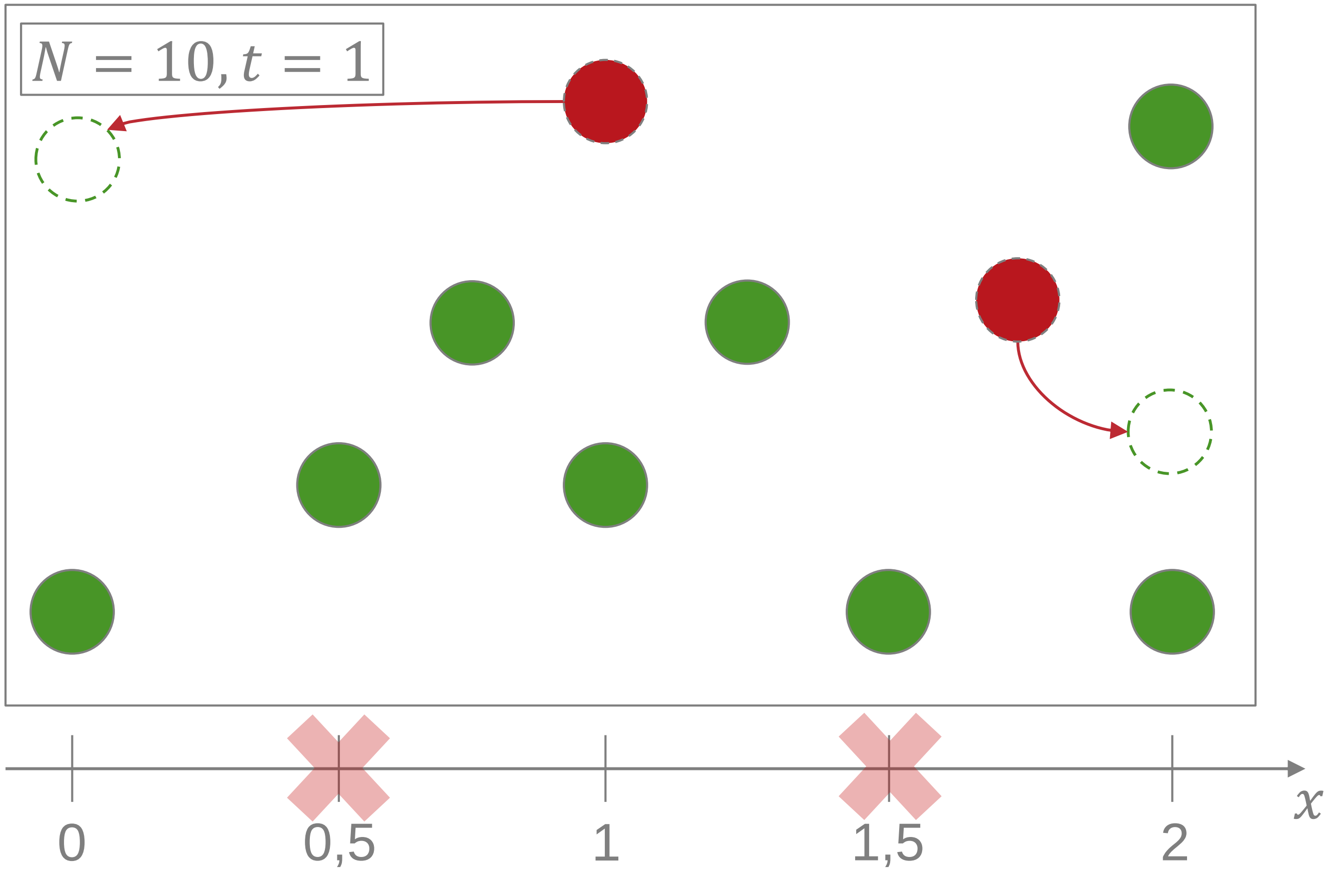
Assume: Self-organised process!

attribute value	frequency
0	2
0,5	2
1	2
1,5	1
2	3

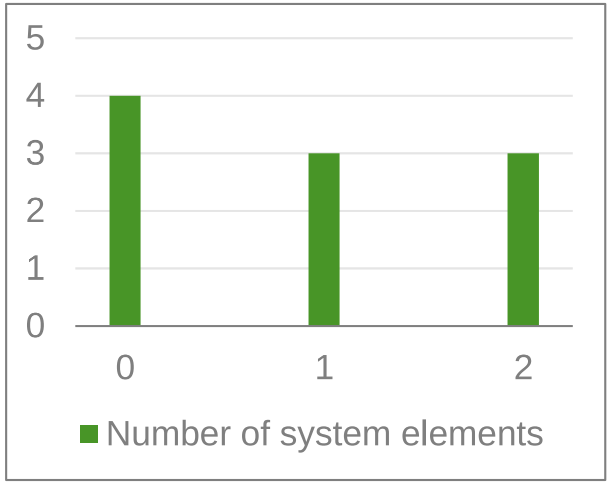


Quantification of abstraction change (3)

Numerical precision changed from *double* to *int*



attribute value	frequency
0	4
~~0,5~~	
1	3
~~1,5~~	
2	3



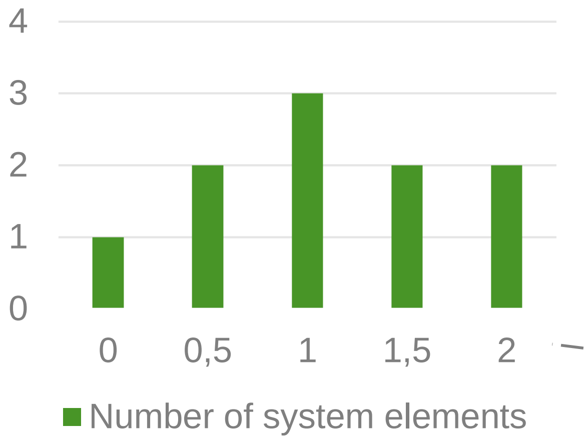
Different state (self-organisation) AND abstraction level

$$H_x^1 = -\left(\frac{4}{10} * ld\,\frac{4}{10} + \frac{3}{10} * ld\,\frac{3}{10} + \frac{3}{10} * ld\,\frac{3}{10}\right)$$

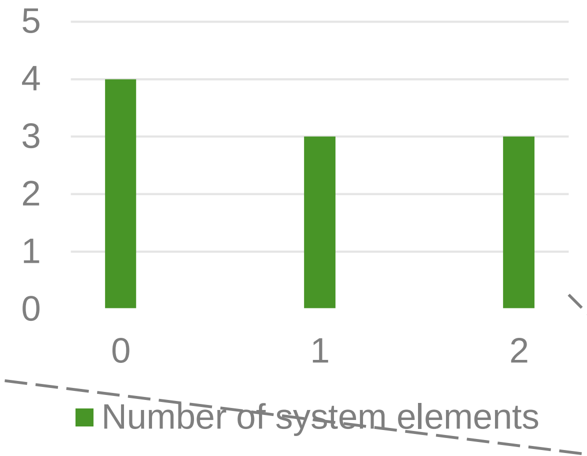
$$H_x^1 = 1{,}57095059445$$

Quantification of abstraction change (4)

What happened? Did order increase?



S at $t = 0$



S at $t = 1$

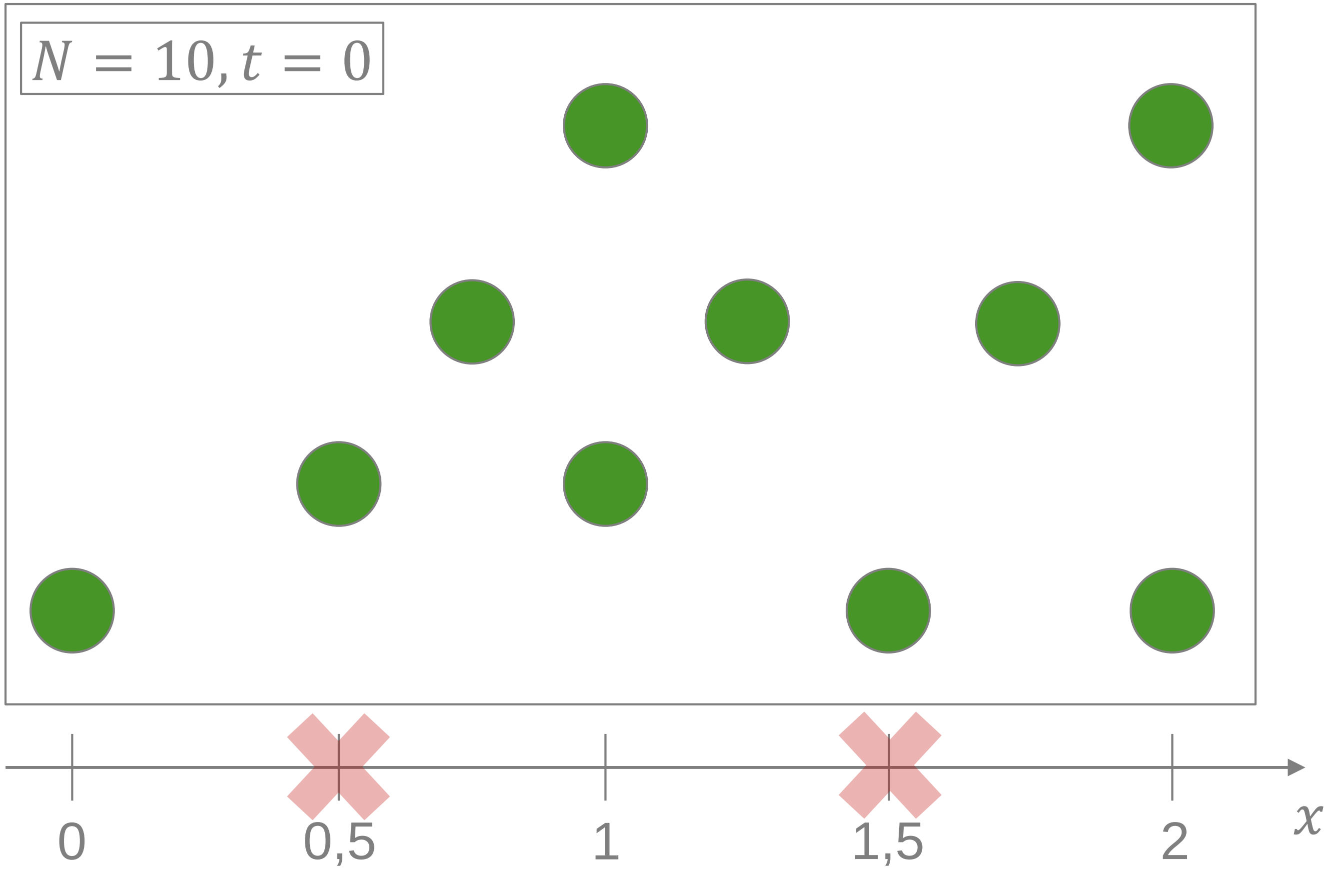
Let's calculate the emergence M:

$$M = H_x^0 - H_x^1$$
$$M = 2{,}24643934467 - 1{,}57095059445$$
$$M = 0{,}67548875022$$

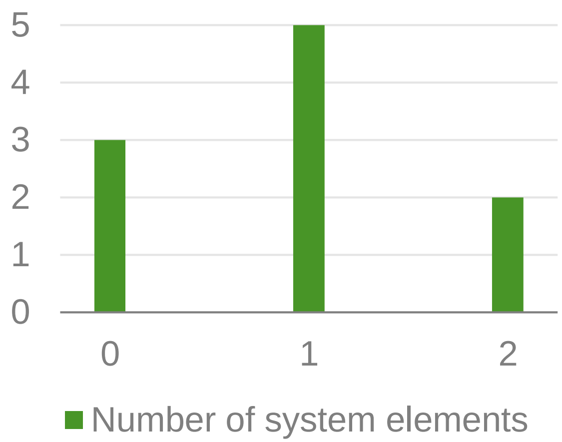
Result: *Increase* in terms of *order* (*decrease* of *entropy*)!
But: Subtract influence of ΔH_{view} (i.e. abstraction change)!

Quantification of abstraction change (5)

First step: adjust the abstraction at time $t = 0$:



attribute value	frequency
0	3
~~0,5~~	
1	5
~~1,5~~	
2	2



Change of abstraction level (double → int)

$$H'^{0}_{x} = -\left(\frac{3}{10} * ld\,\frac{3}{10} + \frac{5}{10} * ld\,\frac{5}{10} + \frac{2}{10} * ld\,\frac{2}{10}\right)$$

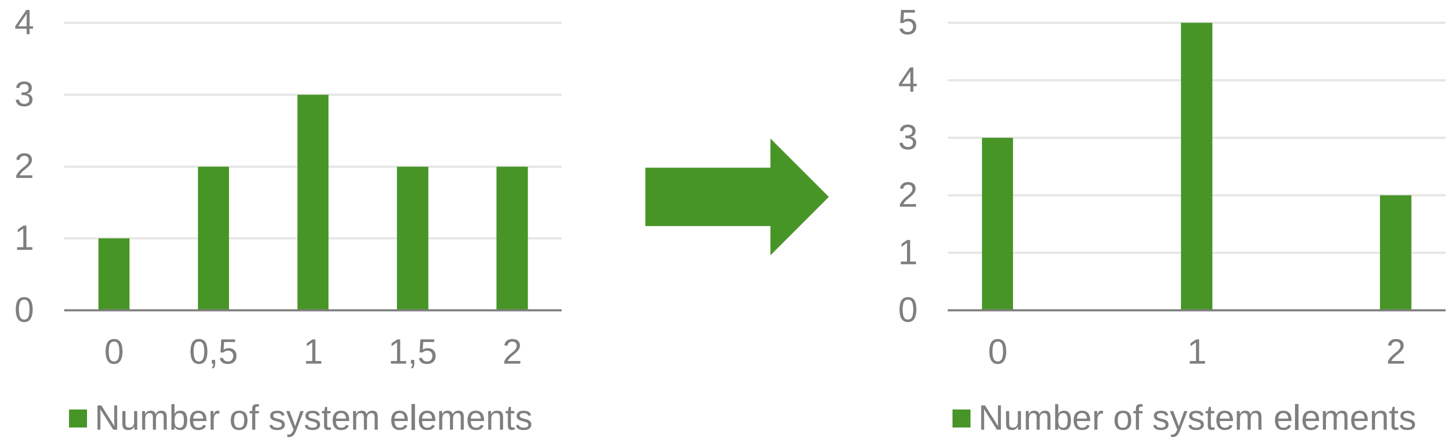
$$H'^{0}_{x} = 1{,}48547529723$$

Quantification of ΔH_{view}

Now, ΔH_{view} calculates:

- $\Delta H_{view} = H_x^0 - H_x'^0$
- $\Delta H_{view} = 2{,}24643934467 - 1{,}48547529723$
- $\Delta H_{view} = 0{,}76096404744$

- Thus, we have an *increase* in terms of *order*, caused by the abstraction level change (less uniformly distributed)!



S at $t = 0$ S at $t = 0$

Quantification of ΔH_{view} (2)

- If we compare the states at $t = 0$ and $t = 1$ on the same level of abstraction, we see a *decrease of order*, due to a *higher degree* of *uniform distribution*!

S at $t = 0$ $\qquad$ S at $t = 1$

Quantification of ΔH_{view} (3)

Final step: subtract ΔH_{view} from the emergence M calculated before:

- $M = H_x^0 - H_x^1 - \Delta H_{view}$
- $M = 2{,}24643934467 - 1{,}57095059445 - 0{,}76096404744$
- $M = -0{,}08547529722$

- And. Ta-da!

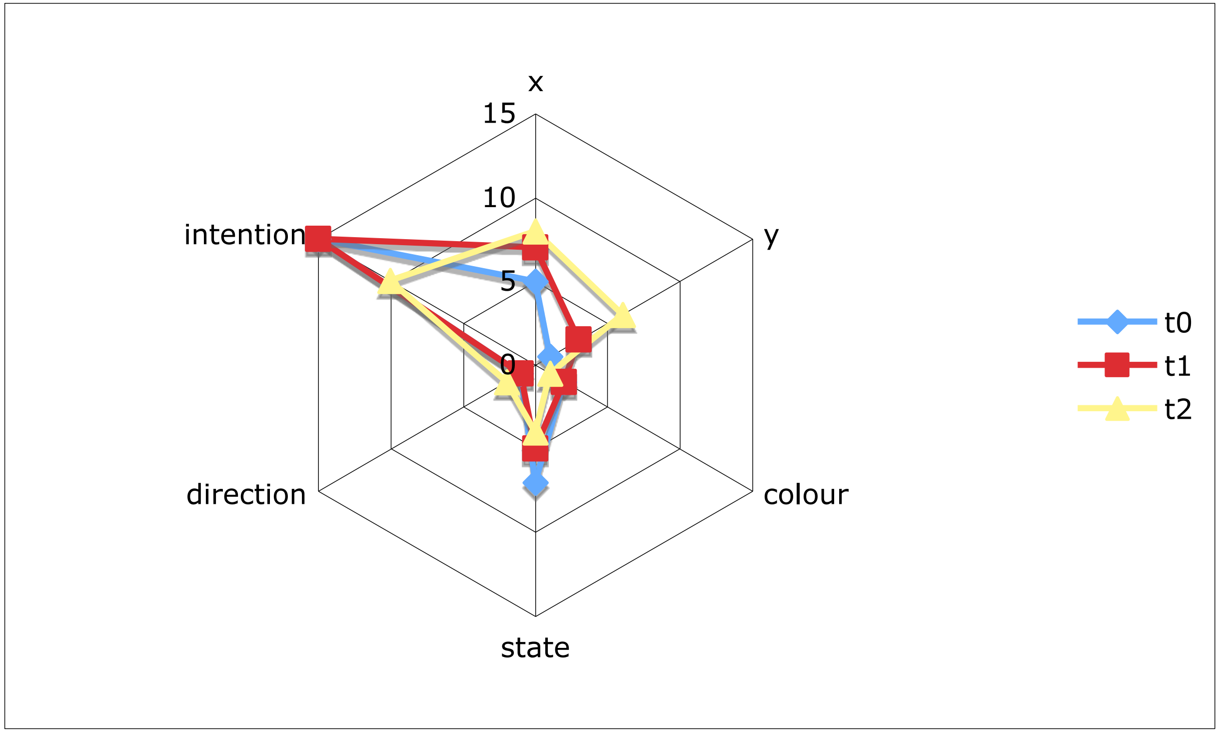
- Now, we get a negative emergence value M which indicates an *increase* in terms of *entropy* from time $t = 0$ to $t = 1$, given the change in the level of abstraction.

How to utilise emergence information?

- Emergence can be calculated for a given system for different attributes.
- It can be used as an early indicator of (emergent) ordering processes.
- System emergence (the total of all attribute emergence values) is not selective enough.
- More interesting: Emergence fingerprint for all relevant attributes.
- Open questions:
  - Which attributes are relevant?
  - What is positive (wanted) and negative (unwanted) emergence?
  - How can we identify results of self-organised processes?

Emergence fingerprint

- Emergence fingerprint = visualisation of all (relevant) attribute emergence values of a system.
  - Visualisation as *n*-dimensional Kiviat graph
- Example
  - x-position, y-position, colour, state, direction, intention

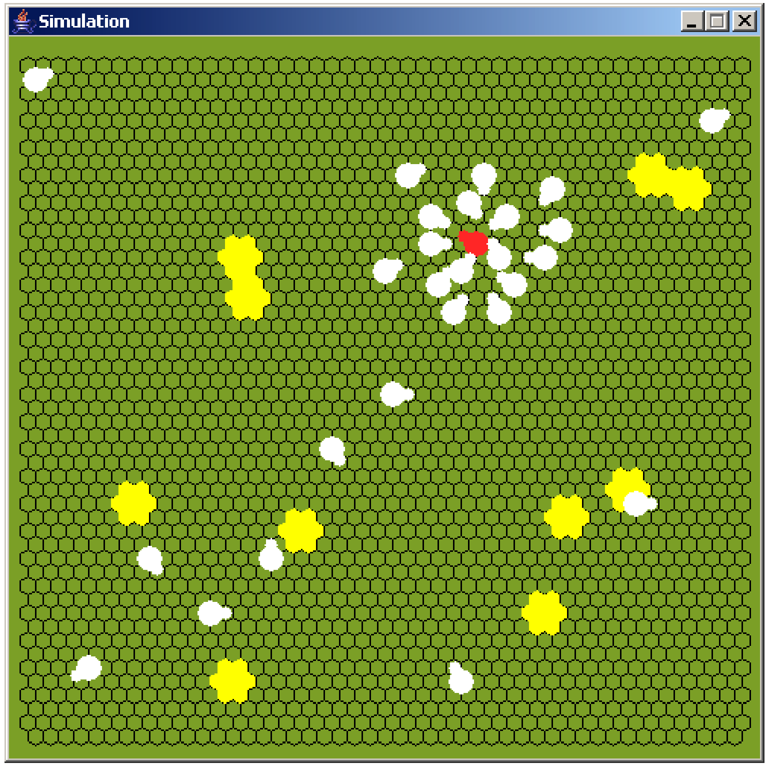


Example: cannibalistic behaviour of chicken

ORGANIC COMPUTING

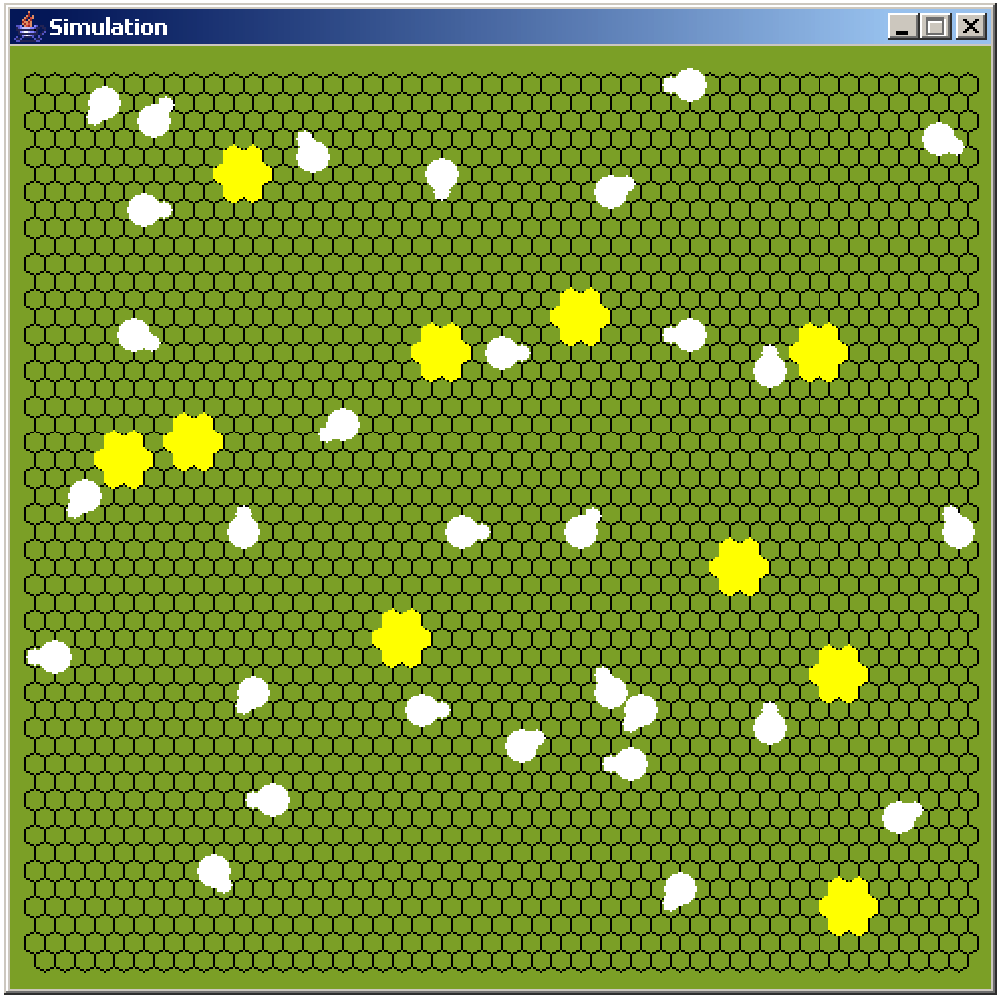
Problem

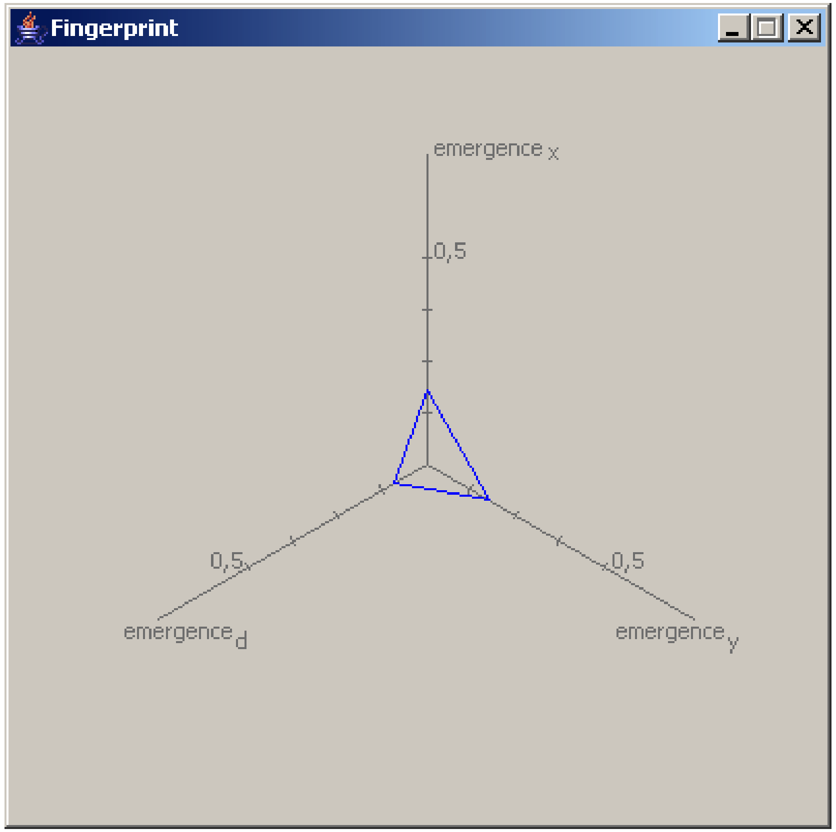
- TiHo Hannover
- Chicken stock in large farms
- Thousands of chicken in one shed
- Slightly injured chicken
  - Other chicken start to attack them
  - … until they die
  - Bad for chicken (→ dead) and owner (→ cost)
- What to do?
  - Noise disturbs chicken, they let up from injured chicken.
  - But: Noise is bad (stress level) ⇒ use it sparingly



Yellow: food source
White: chicken with heading
Red: injured chicken

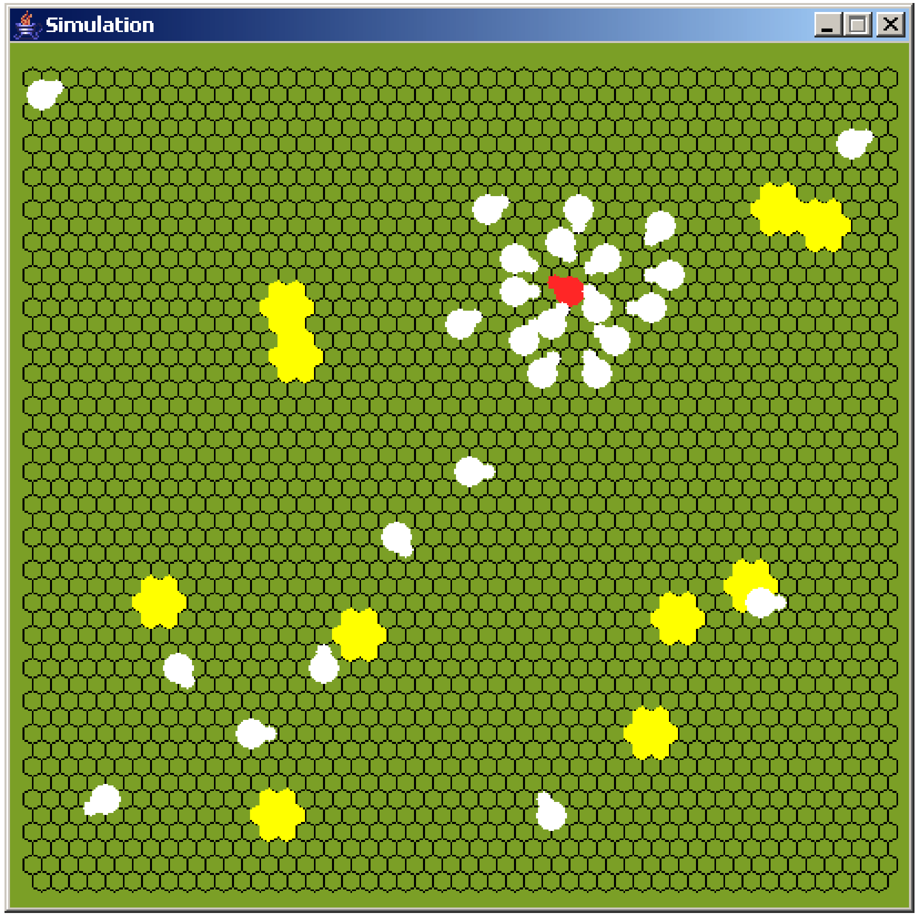
Emergence fingerprint (2)

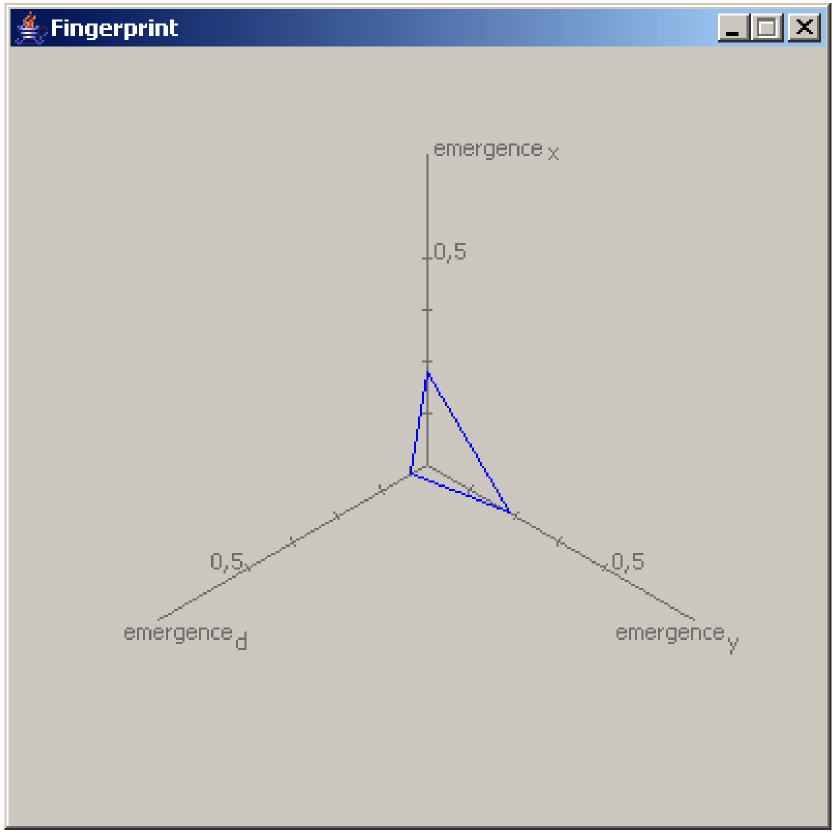




Pattern 1: $M_x = 0.181$, $M_y = 0.177$, $M_{direction} = 0.091$

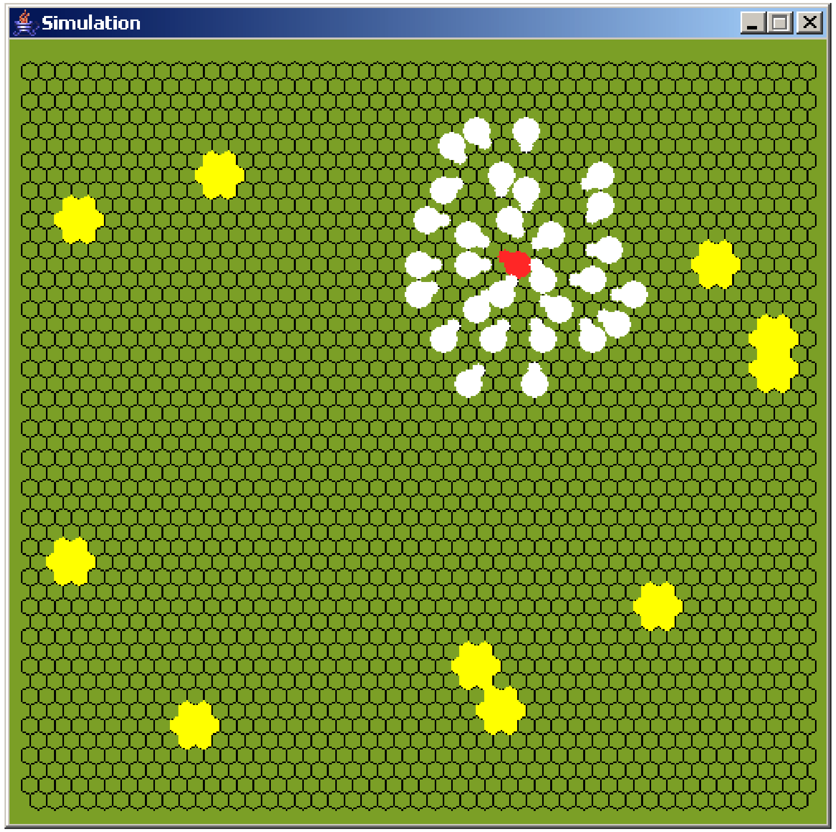
Emergence fingerprint (3)

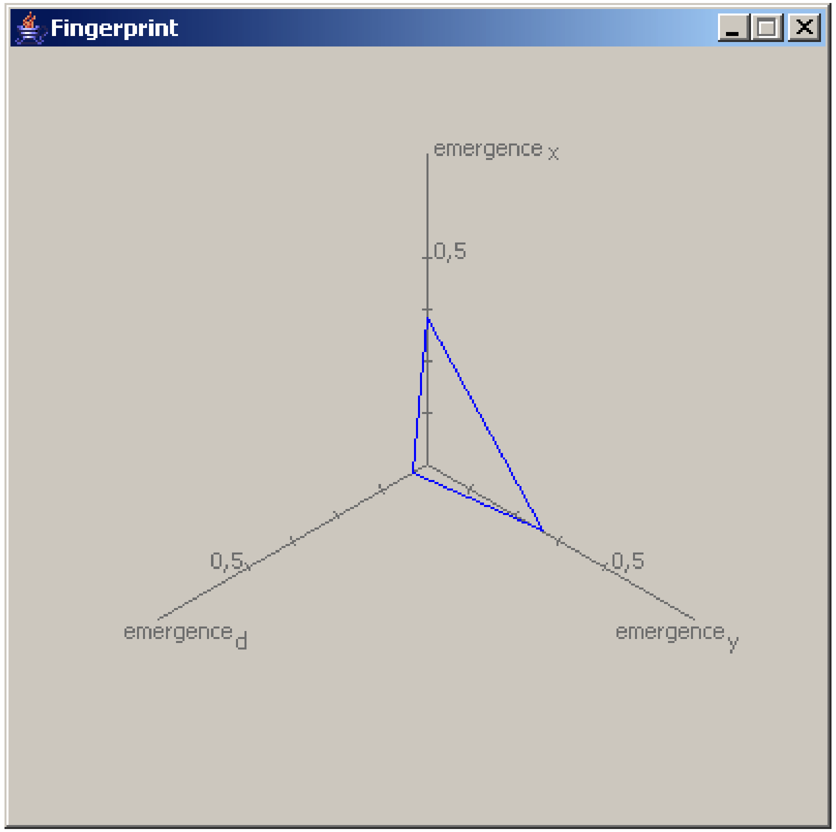




Pattern 2: $M_x = 0.226$, $M_y = 0.237$, $M_{direction} = 0.046$

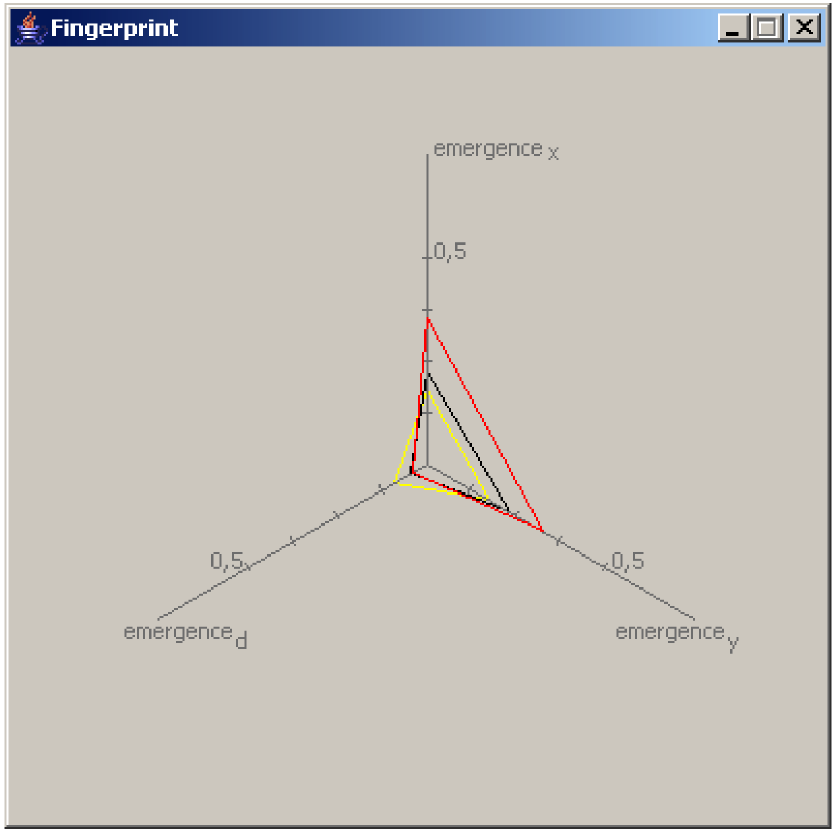
Emergence fingerprint (4)

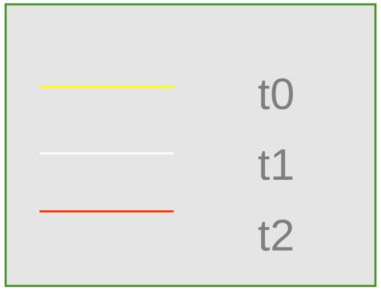




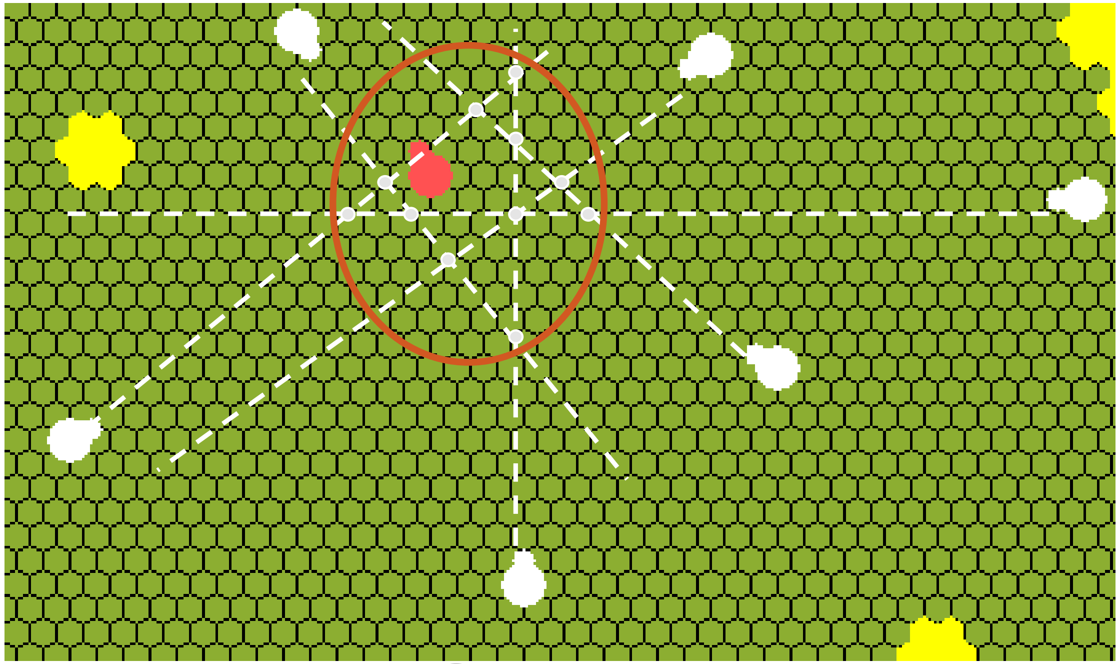
Pattern 3: $M_x = 0.359$, $M_y = 0.328$, $M_{direction} = 0.041$

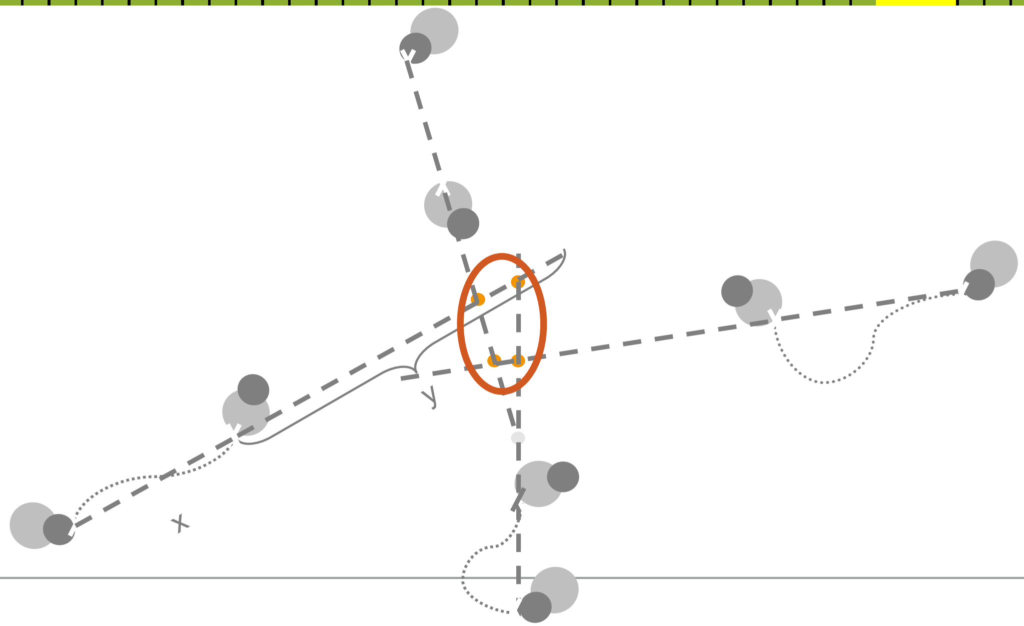
Emergence fingerprint (5)





Emergence fingerprint (6)





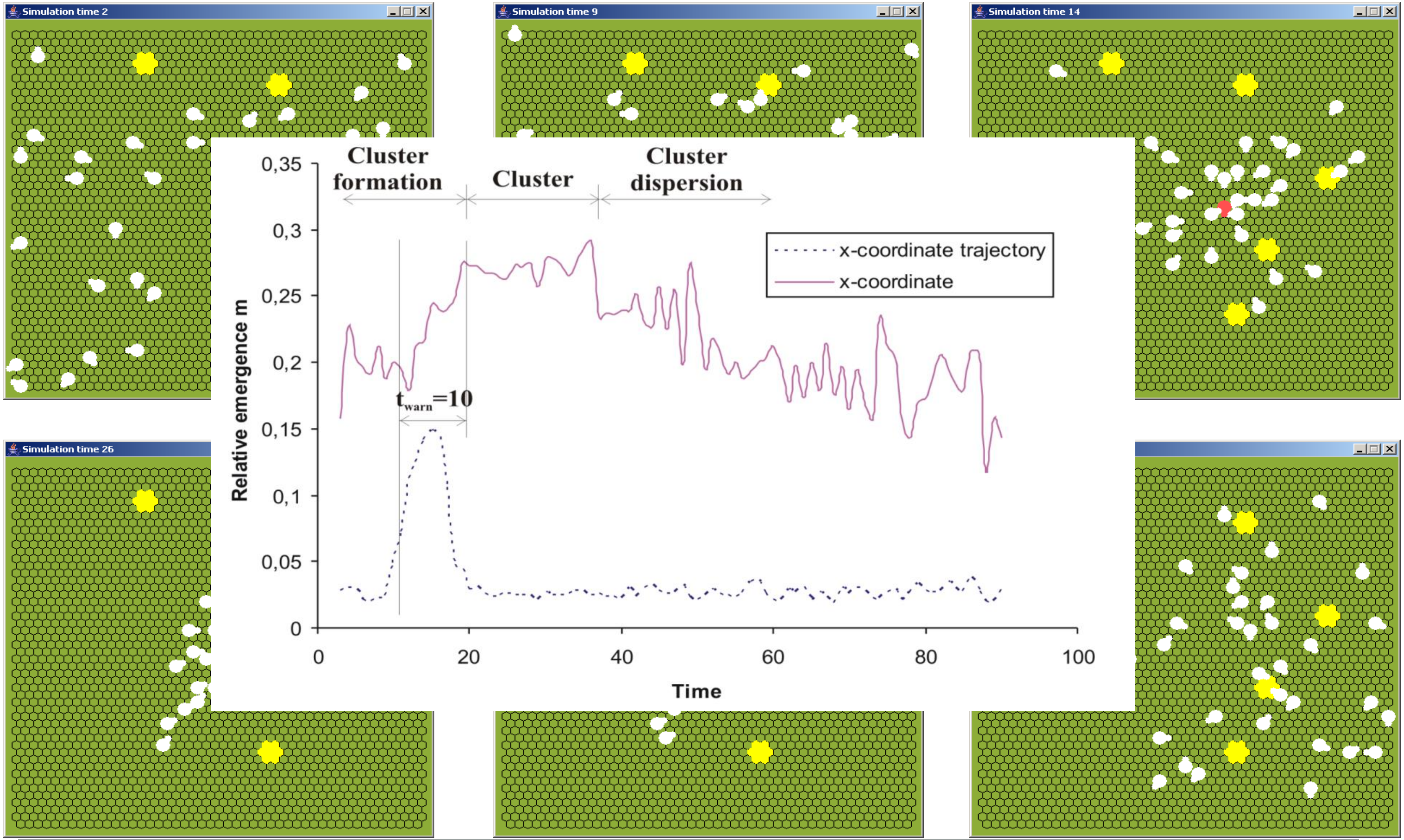
$$v = \frac{x}{\Delta t}$$

$$y = v \times \tau$$

Δt Observation period

τ Prediction period

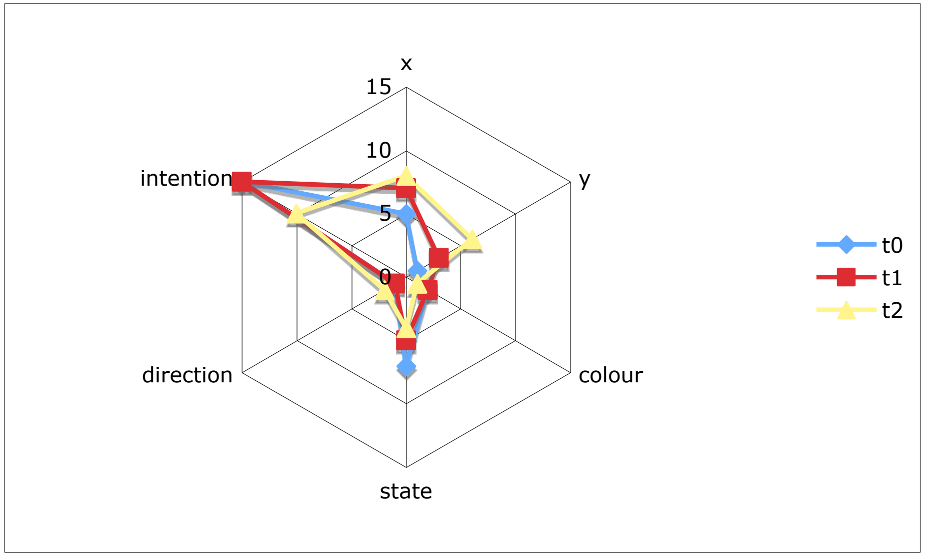
Cluster formation



ORGANIC COMPUTING

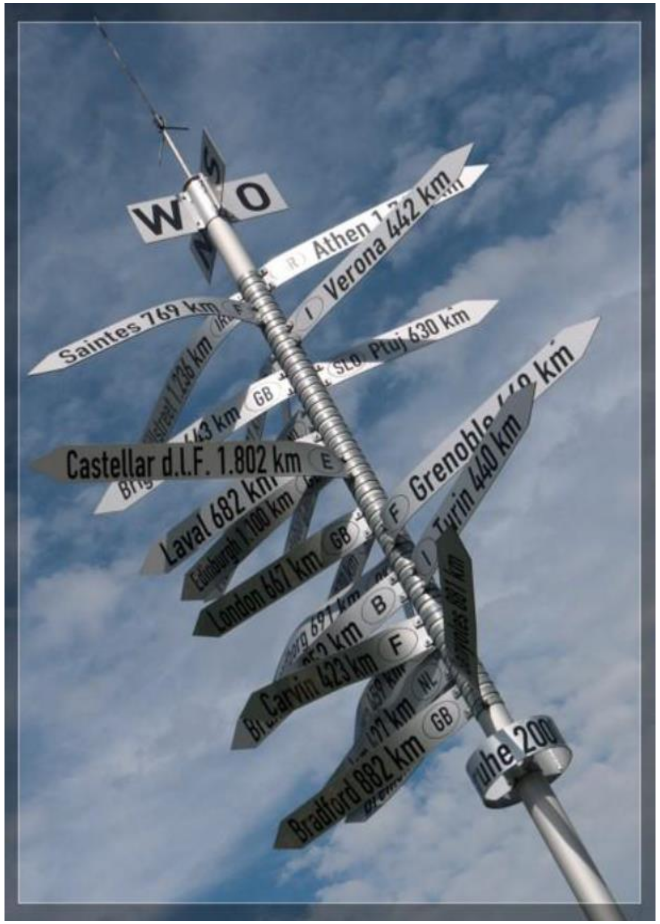
Process

1. Quantify entropy for each attribute.
2. Calculate emergence (M) for each attribute:
   $M = \Delta H = H_{Start} - H_{end} - \Delta H_{view}$
3. ΔH_{view} is the (possible) change of abstraction when observing H_{Start} and H_{end}, e.g. converting *float* to *int* values.
4. a) Determine the system emergence as sum over all attributes.
   b) Illustrate as "fingerprint".
5. Is this due to self-organisation?



Agenda

- A first example: water temples in Bali
- A second example: ants
- Emergence
- Term definition
- Quantification of emergence
- A refined approach to emergence quantification
- Conclusion and further readings



Until now:

- Regarded emergence as the difference between an entropy at the beginning of some process and at the end.
- Discrete entropy difference (DED):

$$DED[x] = H_{start}[x] - H_{end}[x]$$

- A process is called emergent if DED[x] > 0 and the process is self-organised.
- What if we do not know if self-organisation is in place?
- Entropy values are computed for different attributes – which leads to a so-called emergence fingerprint – and this fingerprint serves as basis for certain decisions, e.g., concerning interactions with the system S.

ORGANIC COMPUTING

Approach:

- We want to measure the amount of information we gain when we know that a categorical variable x has value i'.
- In a probabilistic framework: probability $p(x = i')$.
- Another unrelated, categorical attribute y and a value j': $p(y = j')$.
- This information measure has to be additive: If we knew the values of both attributes, the two information values are added.
- Hence: use $-\ln p(x = i')$ and $-\ln p(y = j')$

(which are always non-negative)

- If we observe both values, the amount of information for this observation of statistically independent variables gets:
$$-\ln\left(p(x = i', y = j')\right) = -\ln p(x = i') - \ln p(y = j')$$

ORGANIC COMPUTING

From probabilities to entropy

- We are not interested in specific values of an attribute.
- Instead: We are interested in expected values.
- Hence: determine the expectation of the information with respect to the corresponding distribution.
- This is exactly the entropy, i.e., for a variable x with a corresponding distribution $p(x)$ we get:

$$H[x] = -\sum_{x} p(x) \ln p(x)$$

- Then: sum up over all possible values of x again.
- Entropy describes the expected amount of information which we gain when we observe x.

Limitations of the previous approach

Measure may be unsatisfying in some applications due to:

1. There are many attributes with continuous values in practical applications.
2. Many applications are multi-variate, i.e., based on several (categorical and continuous) attributes.

- The former problem (1) is solved by categorisation of continuous attributes.
  → Could be problematic as entropy measurements depend on size and position of the chosen "bins".
- The latter (2) is solved by analysing the fingerprints.
  → If this analysis is conducted automatically, the different entropy values must be combined at some time.

Approach:

- Multivariate entropy measure for continuous variables.
- Combine all attributes into a vector x.
  Then: continuous entropy (also known as differential entropy) is:

$$H[x] = -\int p(x) \ln p(x)\, dx$$

- where p is the joint density of x.
- p combines all attributes, i.e. several continuous random variables.
- For simplicity: assume that we only have continuous variables.
  → Hybrid (categorical/continuous) approaches are possible.
- Please note: a continuous entropy (in contrast to a categorical one) may have negative values.

Continuous vs. discrete entropy

$$H[x] = -\int p(x) \ln p(x)\, dx$$

$$H[x] = -\sum_{x}^{|Z_x|} p(x) \ln p(x)$$

Density

- Approach relies on estimating the density of a continuous variable.
- Neglect (by now) the functional form of the density function (e.g. to assume that it is Gaussian).
- Then: a non-parametric density estimation approach can be used.
- Assume: given a set X of N observations of x (i.e., samples): $x_0, \dots, x_{N-1}$.
- Goal: estimate $p(x')$ for arbitrary x' (not necessarily $x \in X$).
- Idea: count all samples in a certain environment around x' and divide this number by the size of the environment.

Alternative (smoother):

- Use Parzen window approach, i.e. a kernel density estimator based on a Gaussian kernel:

$$p(x') \approx \frac{1}{N} \sum_{x_n \in X} \frac{1}{(2\pi h^2)^{\frac{D}{2}}} \exp(-\frac{1}{2} \frac{||x' - x_n||^2}{h^2})$$

- where D is the dimensionality of x and h is a user-defined parameter.
- h depends on the data set X – there are a number of heuristics to estimate h (e.g. h is set to the average distance of the ten nearest neighbours from each sample, averaged over the entire data set).

Evaluation of the integral

- Continuous entropy model contains integral.
  → How to evaluate this?
- Remember: data set X contains samples x_n distributed according to p (i.e., $x_n \sim p$).
- Hence: Entropy can be approximated
$$\hat{H}[x] \approx -\frac{1}{N} \sum_{x_n \in X} \ln p(x_n)$$
- where the $p(x_n)$ are estimated using the Parzen approach.
- Note: this discrete approximation of the entropy does not sum up over discrete points in the input space situated on a regular grid.
- Hence: take their non-uniform distribution into account by a correcting factor $\frac{1}{P(x_n)}$.
  → Corresponds to the concept of importance sampling.

From entropy to emergence

- The static approach defines emergence using a difference of entropy values.
  → Emergence is considered as a change of order within a system.
- Here, we define emergence as an unexpected or unpredictable change of the distribution underlying the observed samples.
- Then: use divergence measure to compare two density functions,
  i.e. p(x) at t_0 and $q(x)$ at t_1.
- Possible measure is Kullback-Leibler (KL) divergence $KL(p||q)$.
- Also known as relative entropy.
- Compares two probability density functions.

$$KL(p||q) = -\int p(x)\ln\frac{q(x)}{p(x)}dx$$

Limitations of Kulback-Leibler

- KL divergence is not a true metric since it is not symmetric.
- However:
  - $KL(p||q) \geq 0$ and
  - $KL(p||q) = 0$ only if $q(x) = p(x)$.

- We measure the expected amount of information contained in a new distribution with respect to the original distribution of samples and not with respect to the new distribution:

$$KL(p||q) = -\int p(x) \ln q(x) \, dx + \int p(x) \ln p(x) \, dx$$

- There are concepts for symmetric variants (neglected here).

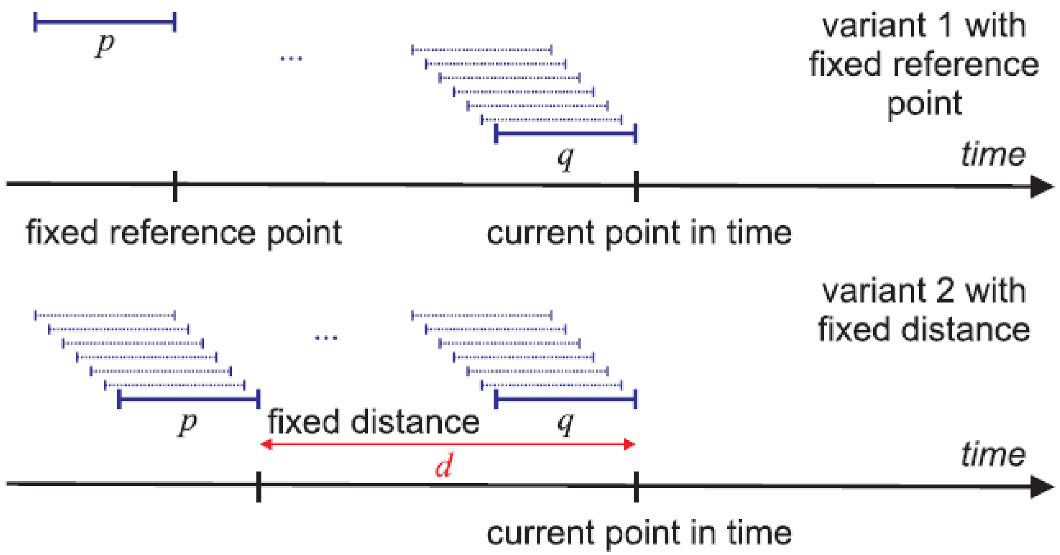
Application of the measures

Measures are applicable to emergence quantification:

- Abstraction from the technical system.
- Consider only distributions of samples in the attribute space.
- Assumption: observation of a number of processes “generating” samples.
- Goal: comparison of the distributions underlying the observed samples.
- Concept: estimation of the distributions at two different points in time, an earlier one (p) and a later one (q).
- Instead of assuming that we get a set of observations at each (discrete) point in time: one single observation at each point in time (these points are considered as equidistant in time).

Sliding window:

- Estimate p and q in sliding data windows.
- Windows have fixed length, must be:
  - long enough to estimate p and q with sufficient reliability.
  - short enough to allow for the assumption that the observed processes are nearly time-invariant in these windows.
- Distinguish:
  - First (earlier) time interval is fixed at a certain point in time, whereas the second interval moves along the time axis with the current point in time.
    → Online application.
  - Both windows move along the time axis in a fixed temporal distance.
    → Distance d is important parameter of the measure-ment technique.



Application of the measures (3)

- Estimation of densities p and q: non-parametric or model-based approaches.
  → Depending on application.
- Hybrid approaches are possible as well.
- If both densities are estimated in a non-parametric approach:
  → Either using the sampling points in the first set of observations ($x_n \sim p$) or those given in the second ($x_n \sim q$).
- Suggestion: Evaluate both intervals and average measures.
  → Get more robust estimates.
- Comparison leads to 'degree' of emergence. In addition:
  - Detection of processes that disappear (i.e., components become obsolete).
  - Detection of newly emerging processes (i.e., new components are required).
    → Novelty detection.
  - Detection of components that change their characteristics (i.e., components change their parameters such as centre or mixing coefficients).
    → Concept drift.

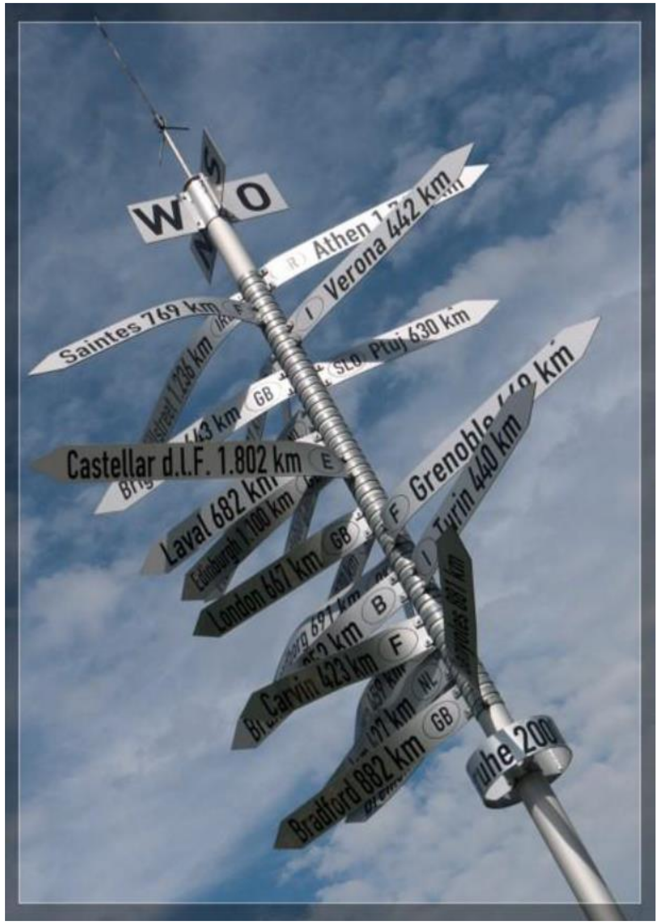
Emergence detection

- Based on probabilistic (or information-theoretic) considerations.
- May be used to determine ‘degree‘ of emergence.
- Contrast to previous approach:
  - Applicable in cases with continuous attributes,
  - Applicable if several attributes have to be combined,
  - Applicable if application allows for model-based density estimates.
- Measures can assess emergence gradually.
- Can further be used to detect novel situations or phenomena such as concept drift.
- In organic systems:
  - Monitor the overall distribution by combining measures for different components.
  - Supervise components individually.

Agenda

ORGANIC COMPUTING

- A first example: water temples in Bali
- A second example: ants
- Emergence
- Term definition
- Quantification of emergence
- A refined approach to emergence quantification
- Conclusion and further readings



From self-organised order to emergence

- Nature as inspiration: Complexity is handled by self-organised order.
- Order is observer- and goal-dependent!
- Self-organised order consists of purposeful self-organisation processes and additional emergent phenomena.
- Same ingredients in organic systems → same processes expected!
- Consequence: We have to measure and master emergence.
- Approach:
  - Observe behaviour of system
  - Measure order (i.e. based on entropy)
  - Compare measures at different points.

Conclusion (2)

This chapter:

- Demonstrated how self-organised order appears in natural, technical and social systems.
- Highlighted the control of complexity by self-organisation and emergence.
- Defined the term 'emergence' and its relation to self-organisation.
- Explained how emergence is quantification for systems with discrete attributes.
- Refined this quantification concept to be applicable to continuous attributes and their combinations.

By now, students should be able to:

- Explain the relation between self-organisation and emergence.
- Briefly summarise the term emergence.
- Give examples for emergent phenomena, e.g. in nature.
- Quantify emergence in technical systems based on discrete attributes.
- Outline how emergence detection is done for systems with continuous attributes.

Further readings

- Steven Johnson: „Emergence – The connected lives of ants, brains, cities, and software“, Scribner publishers, New York, 2001.
- Nelson Fernandez, Carlos Maldonado, Carlos Gershenson: „Information Measures of Complexity, Emergence, Self-organisation, Homeostasis, and Autopoiesis“, online available at: http://arxiv.org/pdf/1304.1842v1.
- Moez Mnif and Christian Müller-Schloer: “Quantitative Emergence”, in: “Organic Computing - A Paradigm Shift for Complex Systems, pages 39 - 52, 2011, Birkhäuser Verlag, Basel, CH. DOI: 10.1007/978-3-0348-0130-0_2
- Dominik Fisch, Martin Jänicke, Bernhard Sick and Christian Müller-Schloer, "Quantitative Emergence - A Refined Approach Based on Divergence Measures," 2010 Fourth IEEE International Conference on Self-Adaptive and Self-Organizing Systems, Budapest, 2010, pp. 94-103. DOI: 10.1109/SASO.2010.31
- Deborah Johnson: “Ants At Work: How An Insect Society Is Organised“. Free Press 2011, New York (USA) and London (UK), ISBN: 978-1451665703.

Questions ...?