Complex Systems Tutorial

Jan Burian
burianj (at) vse.cz

You can find here:

- Basic introduction to Complex Systems Science and relevant modeling tools
- Many links to web resources and a list of relevant literature

- "Complex systems" (4IZ636), lecture on University of Economics, Prague

---

Content

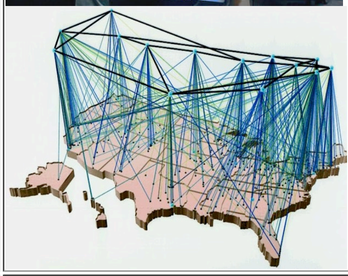
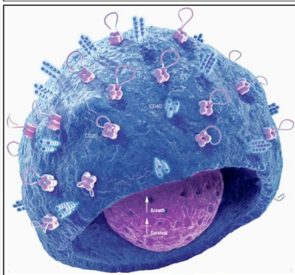
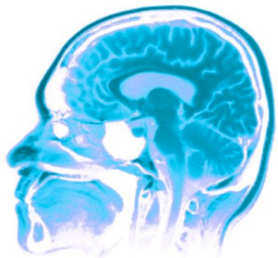
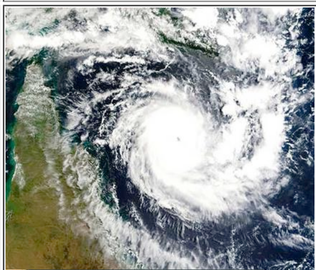
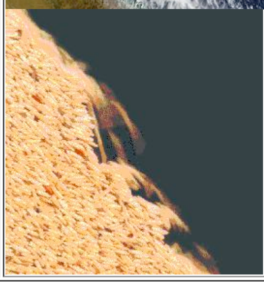
Intuitive Definitions of Complex Systems

System is an entity in terms of **parts** and **relations** between them.

Complex system (*complex* comes from Latin *com-* together + *plectere* to twine or braid) is a system composed from relatively many mutually related parts.

Complex systems are usally (but not always) intricated - hard to describe or understand.

Examples of complex systems

Parts of human society: • Markets • Organizations • Language • Internet	 
Biology: • Cells • Organ – e.g. brain • Immune system • Organisms • Populations • Ecosystem	 
Physics: • Turbulence • Weather • Percolation • Sandpile	 

*The world consists of many **complex systems**, ranging from our own bodies to ecosystems to economic systems. Despite their diversity, complex systems have many **structural and functional features in common** that can be effectively simulated using powerful, user-friendly software. As a result, virtually anyone can explore the nature of complex systems and their **dynamical behavior** under a range of **assumptions and conditions**. (M. Ruth, B Hannon, Dynamic Modeling Series Preface)*

- **Structural relations** define which parts are connected together.
- **Functional relations** define the behavior or dynamics of the system - how does

the change of state of one part influence the state of other connected parts.

Structurally complex system
A system that can be analyzed into many components having relatively many relations among them, so that the behavior of each component can depend on the behavior of many others. (Herbert Simon)

Remember: The number of relationships could be much higher than the number of components!

Dynamically complex system
*A system that involves numerous **interacting agents** whose aggregate behaviors are to be understood. Such aggregate activity is **nonlinear**, hence it cannot simply be derived from summation of individual components behavior. (Jerome Singer)*

System types	Simple structure	Complex structure
Simple dynamics	Example: Pendulum Model: Analytical - differential equations	Example: Closed reservoir of gas Model: Statistical equations
Complex dynamics	Example: Double pendulum Model: Analytical - Complex differential equations or simple simulations Phase portrait of Double pendulum	Example: Ant pile Model: Multi-agent models

Basics of (complex) systems science

Remember: Interconnection of parts matters in complex systems!

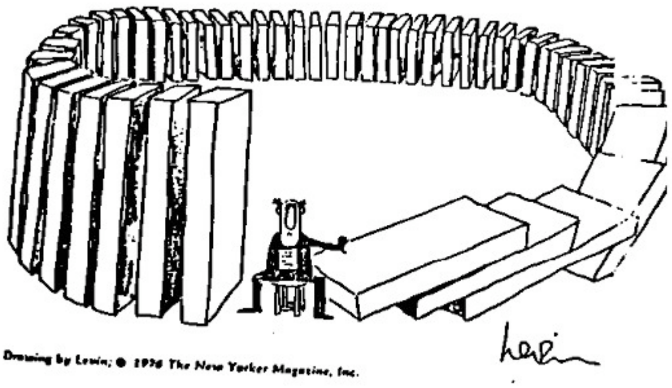
Two complementary approaches to system behaviour

Reductionism: The properites of the whole system could be explained in terms of its parts.	**Holism** : The whole system cannot be determined or explained by its component parts alone. Instead, the system as a whole determines in an important way how the parts behave.

• Understanding of the parts leads to understanding of the whole • Accent on parts	• To understand the whole we must understand also the relations between the parts in the whole system • Accent on relationships

Pragmatic approach rests on a combination of both reductionism and holism.

Feedback



The basic rules of the complex systems could be paradoxically very simple, but their effects are intricate and unexpected because of **feed-back relations** between parts.

Feed-back
The return of a portion of the output of a process or system to the input.

- **Positive feed-back**
  - Deregulative force
  - Drives the system out of equilibrium.
  - Example: Ball on the hilltop, erosion, avalanche, nuclear fission ...
    - Try to find out other examples.

- **Negative feed-back**
  - Regulative force
  - Stabilizes system in the equilibrium.
    - Examples: Ball in the bowl, thermostat, population size and nutrition, classical market ...

- Complex dynamics rests on balance between positive and negative feed-back
  - Examples: Predator-prey relationship, Stock market, Daisy world ..

- Effect of feed-back in complex systems could be counter-intuitive.
  - Example: Building of new highways could lead to more traffic jams, because it initially decreases the waiting times and this increases the desirability of car driving.

System dynamics

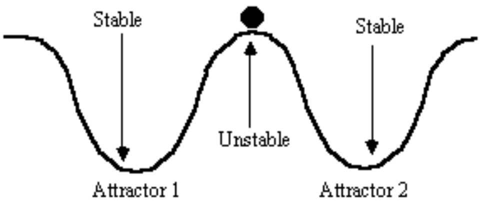
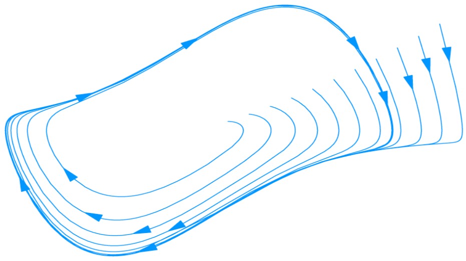
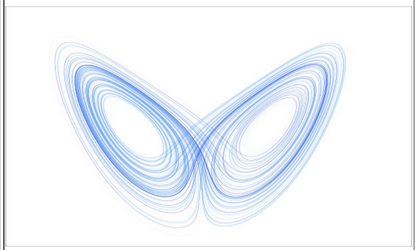
State space *(phase space) is an abstract space in which all possible states of a system are represented, with each possible state of the system corresponding to one unique point in the state space. Dimensions of state space represent all relevant parameters of the system. For example state space of mechanical systems has six dimensions and consists of all possible values of position and momentum variables.*

Dynamics of the system *is the set of functions that encode the movement of* *the system from one point in the state space to another.*

Trajectory of the system *is the sequence of system states.*

Fixed point *is a point in the state space where the system is in equilibrium and does'nt change.*

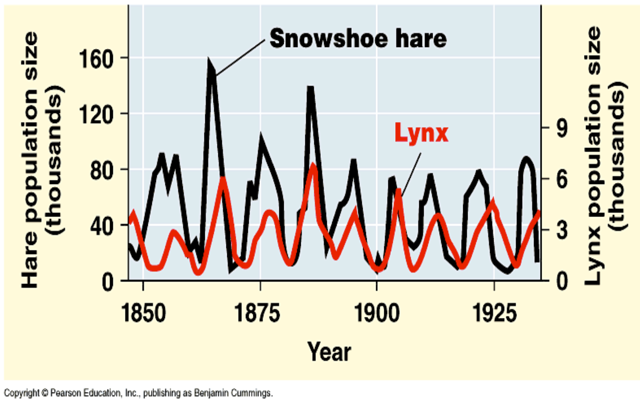
Atractor *is a part of the state space where some trajectories end.*

<table>
<tr><th colspan="3">Attractor types</th></tr>
<tr><td>Fixed point

</td><td>Limit cycle
</td><td>Strange attractor
</td></tr>
</table>

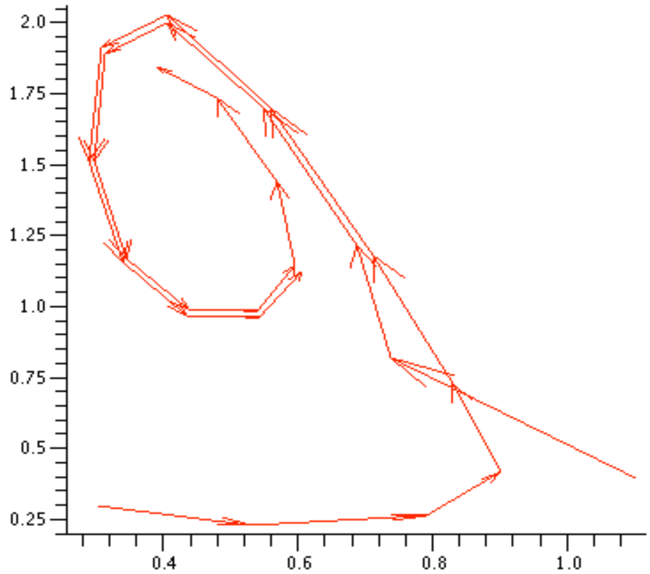
> *Dynamical systems can often be modeled by differential equations dx/dt=v(x), where x(t)=(x1(t), …, xn(t)) is a vector of state variables, t is time, and v(x)=(v1(x), …, vn(x)) is a vector of functions that encode the dynamics. For example, in a chemical reaction, the state variables represent concentrations. The differential equations represent the kinetic rate laws, which usually involve nonlinear functions of the concentrations. Such nonlinear equations are typically impossible to solve analytically, but one can gain qualitative insight by imagining an abstract n-dimensional state space with axes x1, …, xn. As the system evolves, x(t) flows through state space, guided by the 'velocity' field dx/dt = v(x) like a speck carried along in a steady, viscous fluid. Suppose x(t) eventually comes to rest at some point x*. Then the velocity must be zero there, so we call x* a fixed point. It corresponds to an equilibrium state of the physical system being modeled. If all small disturbances away from x* damp out, x* is called a stable fixed point — it acts as an attractor for states in its vicinity. Another long-term possibility is that x(t) flows towards a closed loop and eventually circulates around it forever. Such a loop is called a limit cycle. It represents a self-sustained oscillation of the physical system. A third possibility is that x(t) might settle onto a strange attractor, a set of states on which it wanders forever, never stopping or repeating. Such erratic, aperiodic motion is considered chaotic if two nearby states flow away from each other exponentially fast. Long-term prediction is impossible in a real chaotic system because of this exponential amplification of small uncertainties or measurement errors. (Strogatz, 2001)*

Example: The predator-prey model

We study the dynamics of mutually dependent population size of predators and prey.

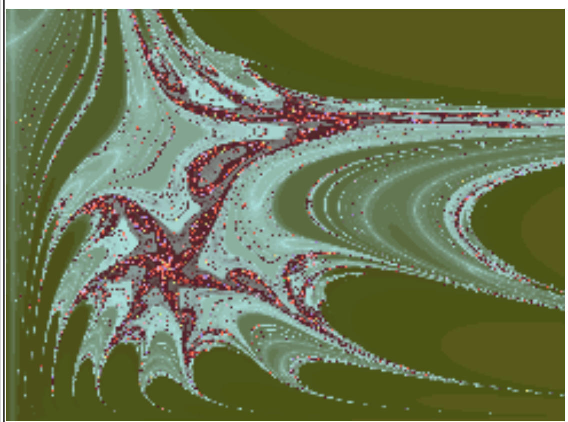


The model is supported by analyses of 100 year fur trapping records of the Hudson's Bay Company.



$$\frac{dx}{dt} = x(\alpha - \beta y) \qquad \frac{dy}{dt} = -y(\gamma - \delta x)$$

Animation of the system dynamics and the two differential equations governing the dynamics (Lotka-Volterra equations). X represents the size of hare population and Y the size of lynx population.



The portrait of dynamics for different initial x and y parameters forms a fractal.
More information about this fractal.

Formal definitions of complexity

Non suitable complexity measures for complex systems

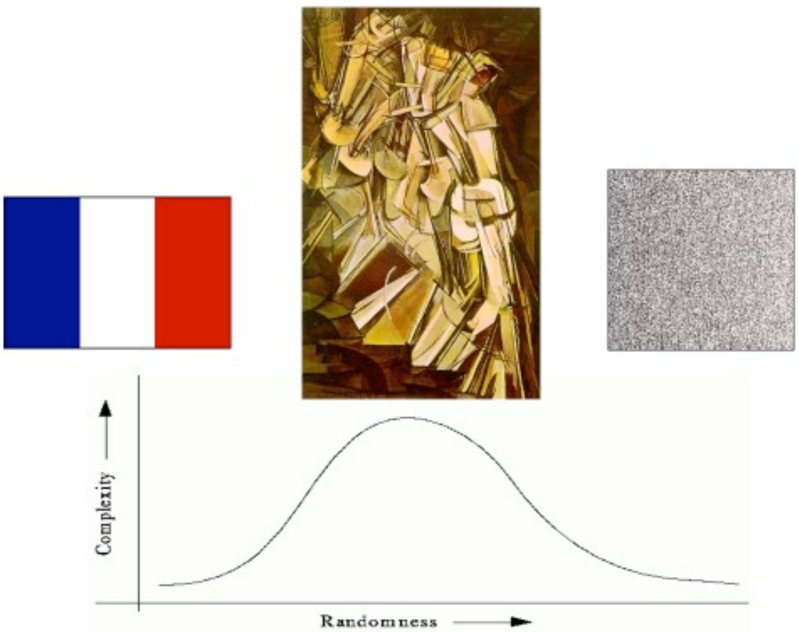
There are many formal definitions of complexity available. Only a small portion of them is suitable for description of complex systems. There are two particular notions of complexity which are not suitable for description of complex systems but have very good sense in other domains.

- **Computational complexity** measures how much time or memory a standard universal computer program needs for solving problems with increasing amount of input data. The dependence of the amount of input data and the required time or memory could be linaer, logarithmic, polynomial, exponential, etc.

- **Algorithmic information content (AIC,** sometimes also called Kolmogorov complexity) of a string of bits is defined as the length of the shortest program that will cause a standard universal computer to print out the string of bits and then halt.
  Gell-Mann (1995) writes „*A random bit string has maximal AIC for its length, since the shortest program that will cause the standard computer to print it out and then halt is just the one that says PRINT followed by the string. This property of AIC, which leads to its being called, on occasion, "algorithmic randomness," reveals the unsuitability of the quantity as a measure of complexity, since the works of Shakespeare have a lower AIC than random gibberish of the same length that would typically be typed by the proverbial roomful of monkeys*.“
  The AIC is called **monotonic** complexity measure because with increasing randomness it also increases.

Suitable complexity measures

Good measures of system complexity should measure the amount of regularities in the system (and not its randomness). Such measures should be low for both very simple systems (where is only one or very few dominant regularities) and random systems (where are almost no regularities). Such measures are called **non-monotonic**. We can say that they are somewhere between order and randomness - on the „**edge of chaos**“ (Langton, 1990).



Neural Complexity

Neural complexity (Sporns et al., 2000, 2002) is a measure inspired by the cognitive processes in the brain. It measures how much the change of activity in one part of he network changes the activity in other parts. The authors described it shortly as a measure of "the difference that makes difference". Neural complexity is one of many complexity measures based on mutual information.

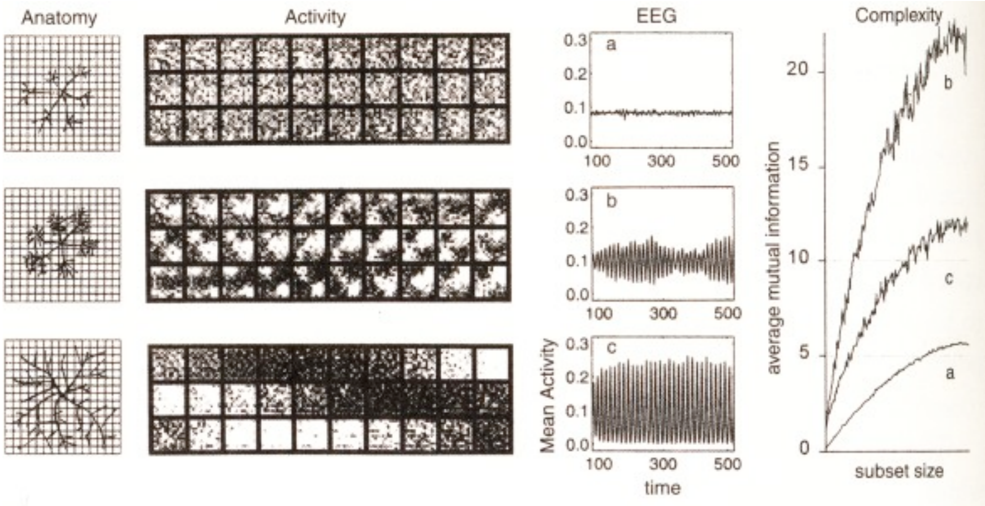
Mutual information between two parts of a system is defined:

$$MI(X_{jk},, X-X_{jk}) = H(X_{jk}) + H(X- X_{jk}) - H(X)$$

There X is the system, X_{jk} is the j-th permutation of a part of size k and X-X_{jk} is the rest of the system.

Neural complexity is formally the sum of average mutual information between subsets of the system and the rest of the system

$$C_N(X) = \Sigma_{k=1..n/2} <MI(X_{ik}, X - X_{ik})>$$



The neural anatomy, neural activity, EEG signal and neural complexity of the brain of an old (a), adult (b) and very young (c) cat. In the old cat there are mostly local specialized connections present but the global integrative connections are missing due to degenrative processes, the result is rather random dynamics. In the adult cat there are both local and global connections present, the result is complex dynamics. In the young cat the local connections are not developed yet but the global connections are already present, the result is regular dynamics (Edelman & Tononi, 2000).

Statistical Complexity

The statistical complexity (Shallizi 2001, 2003, 2004) reflects the intrinsic difficulty of predicting the future states of the system from the system history. It is the amount of information needed about the past of a given point in order to optimally predict its future. Systems with a high degree of local statistical complexity are ones with intricate spatio-temporal organization. Statistical complexity is low both for highly disordered and trivially-ordered systems.

Self-organization and related concepts

Self-organization

> ***Self-organization*** *(First used by Ashby in 1948.). The ability of the system to autonomously (without being guided or managed by an outside source) increase its complexity.*

If a local system is an open system **receiving relatively stable and appropriate amount of energy** from its environment and the local system is composed from **sufficient number of parts** which are able to interact through **positive and negative feedback,** there could (depending on some parameters) be established relatively **stable network of feedback loops**. This process is called self-organization and the established dynamic network is called self-organized system.

Examples of self-organisation:

In Physics:

Benard cells - coherent motion of large number of molecules in heated liquid layers.

Belousov-Zhabotinsky reactions - a specific "coctail" of chemical ingredients loops through visualy discernable states when it receives thermal (heated) or mechanical (stirred) energy.

In Biology:

Flocking - a group of organism can self-organise in a relatively coherent whole which is able to synchronously react on external stimuli.

Ants demo - increasing the length of pheromone trace or the number of ants would lead to self-organization of the food track (result of **stigmergic** interaction between agents and environmnent).

An ecosystem or a whole biosphere - the feedback loops between environment and organism could lead to stabilisation (homeostatis) of some environmental parameters.

Emergence and self-organization

Traditional definition of emergence
*The arising of characteristics of the whole which cannot be attributed to the parts. There arise new **qualitative** and not only **quantitative** changes. Very vaguely: the whole is more than sum of it parts (an statement made already by Aristotle in Metaphysics).*

Modern definition
*Emergence is „the arising of **novel** and **coherent** structures, patterns and properties during the process of self-organization in complex systems." (Corning, 2002; Goldstein, 1999)*

Common characteristics of emergence:

1. Radical novelty (features not previously observed in the system)
2. A global or macro “level” (i.e., there is some property of “wholeness”)
3. Coherence or correlation (integrated wholes that maintain themselves over some period of time)
4. It is the product of a dynamical process (it evolves)
5. It is **meaningfull** for us (i.e. has some pragmatic value for us – we can use it).

Weak emergence: new properties arising in systems as a result of the interactions at an elemental level. The causal conection between the interactions of the parts and the properties of the whole can be traced in great detail.

- Examples:
  - Physics: Temperature, Liquidity (surface tension, friction)
  - Biology: Different levels of organization in living organism (atoms, molecules, cells, organs, organisms, societies, species, ecosystems)
  - Social sciences: Language, culture, market

    In many cases the relationship between parts and the whole depends on large scales of space and time.

Domain	Elementary level	Global level
Geography	Flow of water	Shape of the river bed
Brain	Neuronal firing	Synaptical changes
Organism	Behaviour in specific situations	Ontogeny
Evolution	Life of an individual	Phylogeny
Language	Speech acts	Development of language

Economy	Activity of micro-economical subjects	Macro-economical properties

Strong emergence: the properties of the whole supervene on the properties of the parts. Supervenience describes causal dependence between sets of properties. If property B is causaly dependent on property A, it means that one state of property B could be caused by many states of property A, but one state of property A causes exactly one state of property B.

- Examples:
  - The relationship between physical body and conscious experience (e.g. Mind-body or psycho-physical problem)

Problems

- Some authors disagree with the above definition of emergence (De Wolf, 2005). Can there be emergence without self-organzation and vice versa?
- What is the relation between quality and quantity? When does a new quality arise?

Synergy

The combined (cooperative) effects that are produced by two or more particles, elements, parts or organisms – effects that are not otherwise attainable. (Corning, 2002)

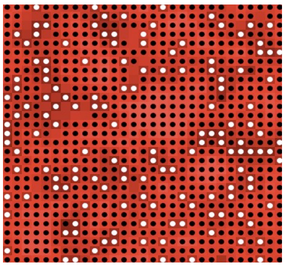
Example: Lichen and other symbiotic organisms

Adaptability

Adaptability is the ability of a system to maintain its complexity in changing environment. Often we can find a feed-back between system and its environment.

System types	**Constructed**	**Self-organized**
Non-adaptable	Example: Classical machines	Example: Crystals
Adaptable	Example: Adaptable robots	Example: Living organisms

Daisy world – an example of self-organized adaptive system



- Aim: Proof of an biological hypothesis that biosphere as a whole can regulate its own environment on a global scale. This hypothesis was proposed by James Lovelock in the early 1970s and he called it the Gaia hypothesis. (Gaia is the name of the Goddess of Earth in ancient Greek mythology.)
- Parts: black and white daisy flowers, environment
- Structure of the environment: 2D matrix
- Functions:
  - Black daisies increase and white daisies decrease the temperature of their environment
  - The reproduction of daisies depends on the temperature of the environment (feedback)
  - The whole system could adapt to external changes of temperature.

Models:
[NetLogo Daisy world model]
[Generalized model at NANIA]

Cellular automatons (CA)

History

- 1940 - 1960-ties: von Neumann and Ulam formalized CA
- 1970: An article in Scientific American (Gardner) about 2D CA called Life (designed by Conway) provoked new interest in CA
- 1983 Wolfram published article about different classes of behavior in CA

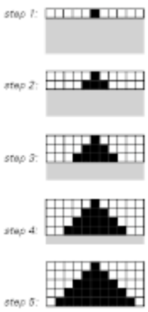
Characteristics

- Bottom-up approach - simulation of **very simple parts** (cells) interacting through **very simple rules** in a **homogenous and regular environment** (matrix) could lead to extremely complex behavior of the whole system
- Precursor of Agent-Based models (ABM)

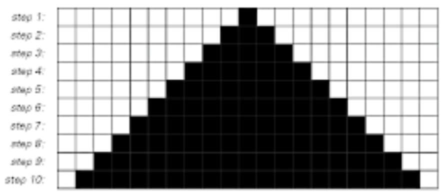
- Discrete space and time
- Discrete states - every cell has finite number of states, this number is the same for all cells
- Discrete dynamics - states of cells change synchronously
- Local interaction in a homogenous spatial matrix - states of the cells are dependent on states of neighboring cell

1D cellular automatons

Basic description of a simple cellular automaton (CA) as presented in (Wolfram, 2002). This book is available on internet and represents a good introduction to cellular automatons (and other simple automatons with complex behavior) but it also gained bad reputation due its egocentric tone.

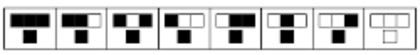


A visual representation of the behavior of a cellular automaton, with each row of cells corresponding to one step. At the first step the cell in the center is black and all other cells are white. Then on each successive step, a particular cell is made black whenever it or either of its neighbors were black on the step before. As the picture shows, this leads to a simple expanding pattern uniformly filled with black.



The cellular automaton consists of a line of cells, each colored either black or white. At every step there is then a definite rule that determines the color of a given cell from the color of that cell and its immediate left and right neighbors on the step before.

For the particular cellular automaton shown here the rule specifies—as in the picture below—that a cell should be black in all cases where it or either of its neighbors were black on the step before.



A representation of the rule for the cellular automaton shown above. The top row in each box gives one of the possible combinations of colors for a cell and its immediate neighbors. The bottom row then specifies what color the center cell should be on the next step in each of these cases. In the numbering scheme described in Chapter 3, this is cellular automaton rule 254.

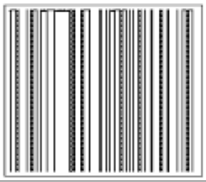
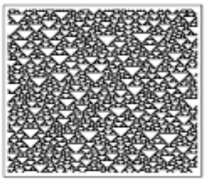
Basic types of rules

- **Elementary** - the state of the cell is dependent on the structure of neighboring cells (as in the above example)
  - If the cells could have k different states and the state of the cell is

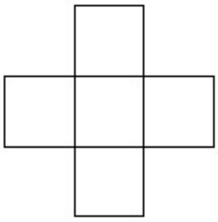
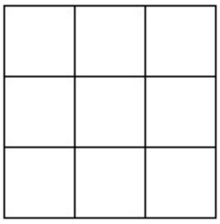
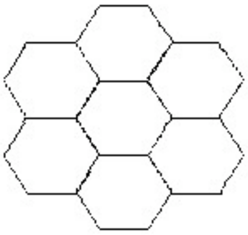
dependent on n neighboring cells, then there exist k^(k^n) rules. For example if each cell could be only in two possible states (the CA is binary) and the state is dependent on three neighboring cells (including itself) then there are 2^(2^3) = 256 rules.
  - From any standard rule you can easily construct a reversible rule.

- **Totalistic** - the state of the cell is dependent on the sum of neighboring cells with specific states (as in the Game of Life - see the next chapter)

Different rules produce very different behavior.

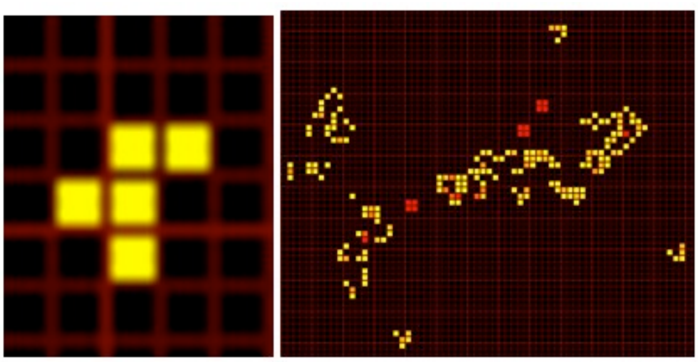
Four classes of behavior in 1D CA (Wolfram, 1983)			
			
Uniform	Repetitive	Random	Complex

2D Cellular automatons and the Game of Life

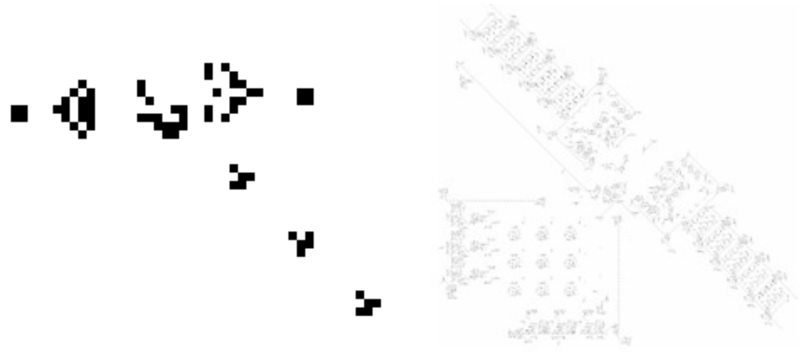
Types of 2D neighborhood		
		
von Neumann	Moore	Hexagonal

Game of Life

- Invented by Conway in 1960s.
- 2D binary CA with Moore neighborhood and totalistic rule:
  1. Any live cell with fewer than two live neighbors dies, as if by loneliness.
  2. Any live cell with more than three live neighbors dies, as if by overcrowding.
  3. Any live cell with two or three live neighbors lives, unchanged, to the next generation.
  4. Any dead cell with exactly three live neighbors comes to life.



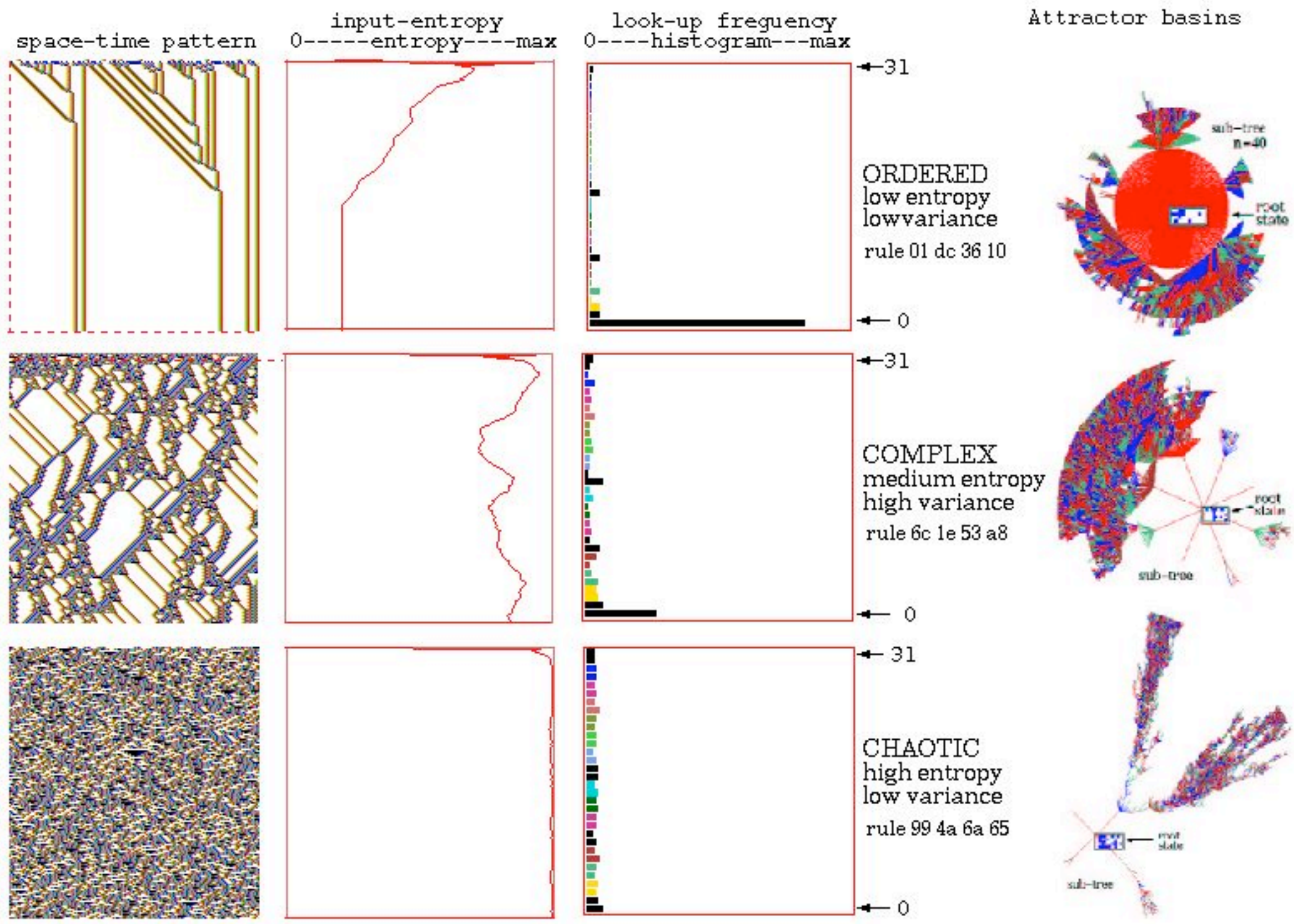
From the five cells on the left (so called F-pentomino) evolved one hundred steps a complex pattern.



Left: An example of an complex oscilator in Life (Gosper's glider gun). Right: Turing machine implemented in Life (Rendell, 2005).

More about CA dynamics

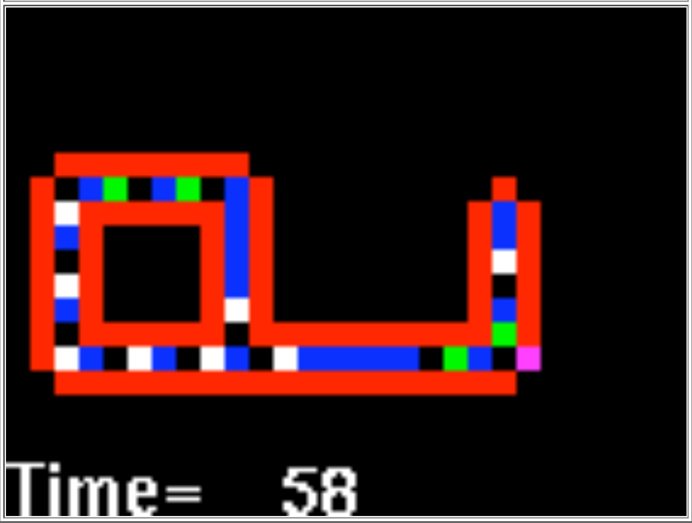
Attractor basins structure and entropy variation in different classes of CA rules (Wuensche, 1998)



Self-replication

Self-replication was first investigated by von Neumann in 1940s. The von Neumann self-reproducing automata is actually a universal constructor' that constructs "any machine" in its 29-state cellular space. In particular, it is capable of Turing universal computation. It solves the self-reproduction problem by reading a tape containing instructions on how to build a copy of itself, provides the copy with a copy of its own input tape, and then presses the ON button starting the copy in operation. In the 1980s, C. Langton and then J. Byl showed that in fact much smaller automata can in fact self-reproduce.

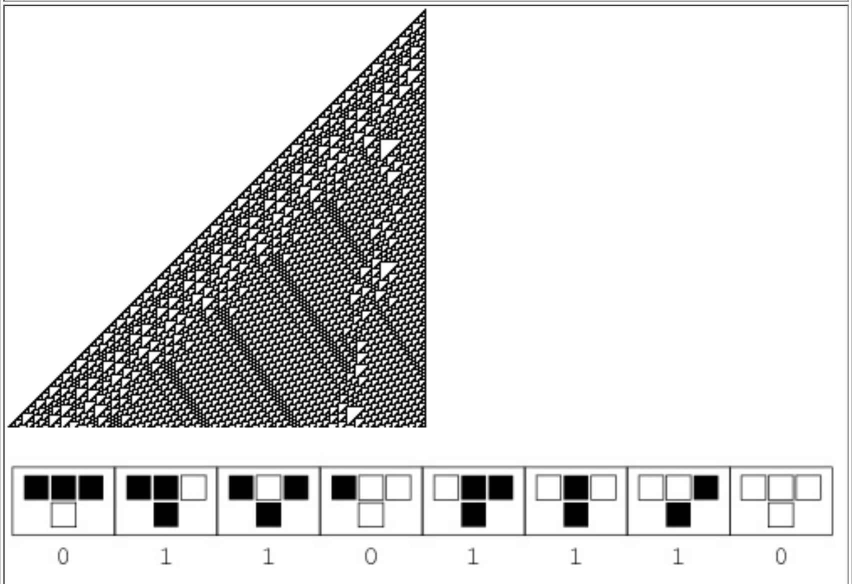
Langton's self-replicating loop



Computational universality of some CAs

- In 1960s von Neumann designed CA emulating an Universal Turing machine
- In 1980s Life was proved to be equivalent to Universal Turing machine
- In 1990s Cook proved the rule 110 in 1D CA to be equivalent to Universal Turing machine

CA rule 110



Generalizations of CAs

- More than two states
- Extended or "Margolus" neighborhoods ([models of gas interaction](#))
- Continuous CA
  - The states can be real numbers in interval <0; 1> .
  - This type of CA could be used for simulation of chemical reactions and diffusion.
- Boolean networks
  - System of N binary-state nodes with K inputs to each node and one of the possible Boolean functions of K inputs
  - Proposed by Kauffman in 1969 as a model of genetic regulatory networks.
- Discrete dynamical networks (DDN), similar to RBN, but allowing a value range greater then 2. CA and RBN are special cases of DDN.

Applications of CA

- Physical models
  - Crystal growth
  - Fluid flow
  - Percolation
  - Reaction-diffusion models
  - Ferromagnetism
- Biology
  - Pattern formations
  - Cellular growth
  - Forrest growth
  - Spread of diseases, species, fire etc.
  - Genetic regulatory networks
- Social sciences
  - Traffic
  - Urban Growth
- Mathematics
  - Cryptography
  - Firing squad problem
  - Majority problem

A seashell with CA-like patterns

Resources about CA on WWW

Mireks's Java Cellebration - one of the best CA tools

Game of Life and Cellular automaton on Wikipedia

CA Tutorial by Alexander Schatten,

"History of Cellular Automata" from Stephen Wolfram's "A New Kind of Science"

General article about Cellular Automata by Cosma Shallizi

The DDLab manual by Andy Wuensche with many information about CA, discrete dynamical networks and their attractor basins.

Complex Networks

In CA there was interaction possible only between neighbouring cells in a spatial matrix. But the interaction between active parts of a system could be generally described by a network where the active components are represented by nodes and the interactions by edges. Complex networks are a subgroup of networks with "interesting" properties. Natural and social networks are often complex.

Motivation for the study of complex networks:

- Complex networks are almost everywhere
- Many complex networks have similar properties
- Structure of interactions affects the resulting dynamics

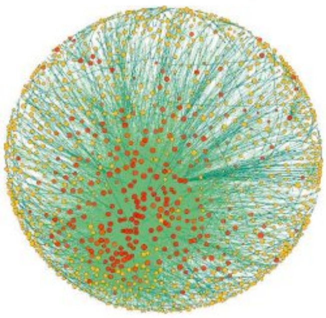
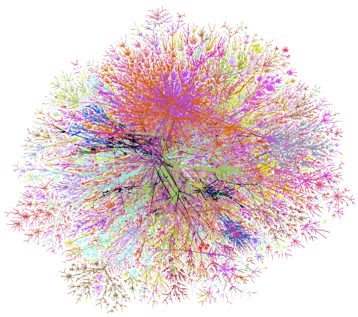
<table>
<tr><th colspan="2">Examples of complex networks</th></tr>
<tr><td>

Human society

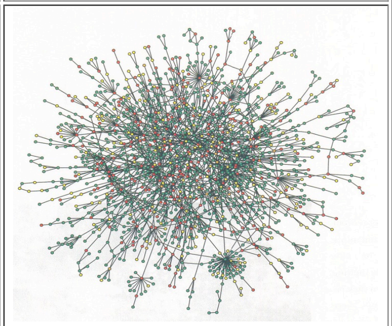
Social networks

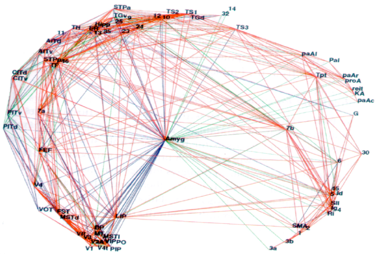
Economics
Epidemiology
Collaboration and Citation networks
Spreading of innovations

Electric grid
Internet

</td><td>

The 1318 transnational corporations that form the core of the economy. (Vitali S. et al., 2011)

Structure of internet (nodes - servers, edges - connections) (Hal Burch and Bill Cheswick, Lumeta Corp.)
</td></tr>
<tr><td>
Biology

Food chains
Gene regulation networks
Metabolism networks

</td><td>

Structure of yeast protein interactions (nodes - proteins, edges - reactions) (Barabasi et. al., 2003)
</td></tr>
</table>

• Neural networks	 Macaque cortex network (Young, 1993)

For more examples see [Gallery of network images]

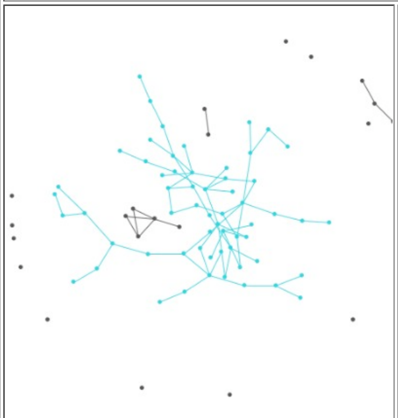
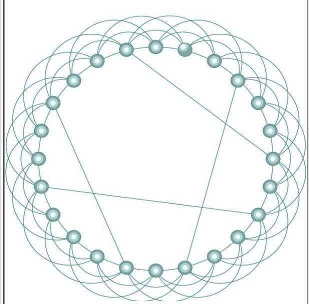
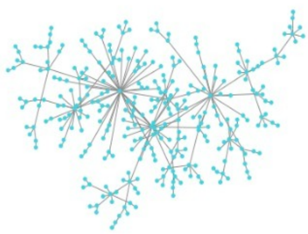
Basic notions of Graph theory

- Graph *G = (V, E)*
- *V* is the set of vertices (or nodes)
- *E* is the set of oriented or not oriented edges (or links)
- Path length is the minimal number of vertices between two vertices
- Degree of a vertex k_i is the number of edges connected with the node v_i
- Degree distribution *P(k)* is the distribution of probabilities that a random vertex has a degree *k*
- Clustering coefficient for an undirected graph:
  $$C_i = \frac{2|\{e_{jk}\}|}{k_i(k_i-1)} : v_j, v_k \in N_i, e_{jk} \in E.$$
  - The clustering coefficient C_i for a vertex v_i is given by the **number of links between the vertices within its neighbourhood (e_{jk}) divided by the number of links that could possibly exist between them** (In a directed graph e_{jk} is distinct from e_{kj}, and therefore for each neighbourhood N_i there are $k_i(k_i - 1)$ links that could exist among the vertices within the neighbourhood (k_i is the total (in + out) degree of the vertex). In undirected graphs e_{ij} and e_{ji} are considered identical. Therefore, if a vertex v_i has k_i neighbours, $k_i(k_i - 1)/2$ edges could exist among the vertices within the neighbourhood.

Some properties of complex networks:

- Short average path between vertices
- Specific degree distribution
- High clustering coefficient (in small world networks and some scale-free networks)

Types of complex networks

Types of complex networks (Huang, 2005)		
 Random network	 Small World network	 Scale-free network

Random networks

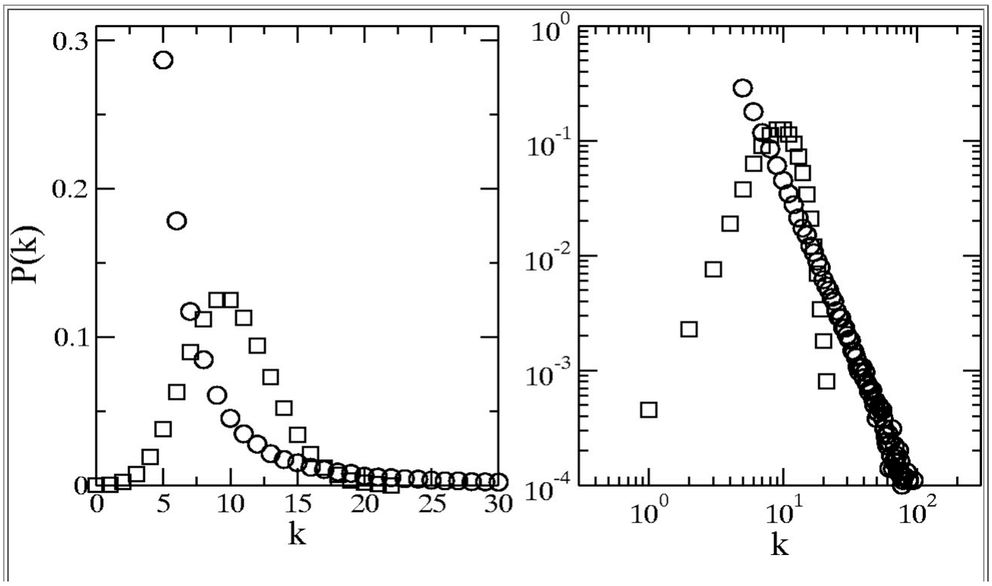
- First models of complex networks
- Developed in 1960s by Erdos and Rényi
- Model
  - Add edges between nodes with probability *p*
- Properties
  - Length of paths close to *log n* (where n is the number of nodes)
  - Small changes of *p* can lead to sudden emergence of new characteristics
    - Giant component (the biggest connected part starts to grow very fast when *p* comes to *1/n*)
  - Poisson degree distribution
    - Different from most real complex networks
  - Clustering coefficient close to *p*
    - Lower then in most real complex networks

Small world networks

- Model of human society
  - According to Milgram´s experiments in 1967 there are only six people between any two people in the network of people personally knowing each other (six degrees of separation)
- High regularity with few irregular connections leads to low path lengths
- Model
  - Arrange *n* nodes in a circle and connect *k* neighbors together
  - Then add new edges (or rewire the existing ones) with probability *p*
- Properties
  - Length of paths close to *log n*
  - Degree distribution similar to random networks
  - High clustering coefficient

Scale-free networks

- Very common in real complex networks but not omnipresent
- Model
  - Start with few nodes and edges
  - Add new nodes and connect them with higher probability to nodes with higher degree (the rich gets richer)
  - After a while there will emerge nodes with a very high degree (hubs)
- Properties
  - Short path lengths
  - Scale-free degree distribution $P(k) \sim k^{-\gamma}$



Comparison between the degree distribution of scale-free networks (circle) and random graphs (square) having the same number of nodes and edges. For clarity the same two distributions are plotted both on a linear (left) and logarithmic (right) scale. The bell-shaped degree distribution of random graphs peaks at the average degree and decreases fast for both smaller and larger degrees, indicating that these graphs are statistically homogeneous. By contrast, the degree distribution of the scale-free network follows the power law , which appears as a straight line on a logarithmic plot. The continuously decreasing degree distribution indicates that low-degree nodes have the highest frequencies; however, there is a broad degree range with non-zero abundance of very highly connected nodes (hubs) as well. Note that the nodes in a scale-free network do not fall into two separable classes corresponding to low-degree nodes and hubs, but every degree between these two limits appears with a frequency given by P(k). (Albert, 2005)

The Potential Implications of Scale-Free Networks (Barabasi et al., 2005):

- Computing
  - Computer networks with scale-free architectures, such as the World Wide Web, are highly resistant to accidental failures. But they are very vulnerable to deliberate attacks on hubs.
  - Eradicating viruses, even known ones, from the Internet will be effectively impossible.
- Medicine
  - Vaccination campaigns against serious viruses, such as smallpox, might be most effective if they concentrate on treating hubs--people who have many connections to others. But identifying such individuals can be difficult.
  - Mapping out the networks within the human cell could aid researchers in uncovering and controlling the side effects of drugs. Furthermore, identifying the hub molecules involved in certain diseases could lead to new drugs that would target those hubs.
- Business and politics
  - Understanding how companies, industries and economies are interlinked could help researchers monitor and avoid cascading financial failures.
  - Studying the spread of a contagion on a scale-free network could offer new ways for marketers and politicians to propagate their products and ideas.

Resources about complex networks

Chapter about networks in *Complex Science for a Complex World*.

Exploring complex networks, an article by Strogatz in *Nature*.

Scale-free networks, an article by Barabasi and Bonabeau in *Scientific American*.

NetLogo models:

- Giant component in random networks
- Evolution of small world network
- Preferential attachment model of scale-free network

Very short introduction to modeling methodology

> *Models are always "wrong" but sometimes could be useful!* (Georg E. P. Box)

- All models are abstracted and simplified (like a map of a landscape).
  - We describe only those parts of reality which are important for us and only to the extent allowed by our technical limits.
  - Useful models could help us to get insight into the structure and behaviour of reality.
  - Bad models don't tell us anything new and only waste our time (that's the better alternative) or can lead to bad prediction about the reality

- Logic and modeling:
  - **Deduction** – we know the principles and try to predict the system behavior. (Example: How will the electricity market behave during the year?)
  - **Induction** - we know the behavior and search for underlying fundamental principles of system dynamics. Is the model robust? Does the model lead to the same or similar behavior for a large range parameter values? (Example: How does the stability arise in a predator-prey ecosystem?)
  - **Abduction** – we search for the best explanation (basic assumptions and parameters) of specific interesting results. (Example: When will the stock market tend to crash?)

*A **different look at logical relationship** between a multiagent model and reality: Axelrod (2003) points out: "like deduction model starts with a set of explicit assumptions. But unlike deduction, it does not prove theorems. Instead, a simulation generates data that can be analyzed inductively". Induction comes at the moment of explaining the behavior of the model. It should be noted that although induction is used to obtain knowledge about the behavior of a given model, the use of a model to obtain knowledge about the behavior of the real world refers to the logical process of abduction. Abduction, also called inference to the best explanation, is amethod of reasoning in which one looks for the hypothesis that would best explain the relevant evidence, as in the case when the observation that the grass is wet allows one to suppose that it rained.* (Encyclopedia of Complexity)

Steps of modeling:

- What exactly is our problem and what do we want to achieve with the model?
- Do we need a model at all?
- Are there already similar models?

- Choosing the scale
  - Space – what is the basic part of the system?
  - Time – what does represent the basic step of the simulation and how far in the future we want to predict the behavior?
- Choosing the aspects of the reality we want to model (abstraction)
  - Extensive boundaries: How many aspects of reality to include in the system
  - Intensive boundaries: How detailed will be the description of these aspects
  - Look for simplicity. Always start with simple models and gradually add new features.
    - Remember: Without simplicity you will get stuck in tons of data but too simple models can lose the connection with reality.

- What are the key parts, processes and parameters of the model?
- Choosing appropriate description and representation of the model
- Choosing the modeling tools

- Verification - check if the model does what we suppose it should do.
- Validation – check if the model behaves in accord to the reality

- "Playing" with the model – repeated executing of the model, changing parameters or other aspects and observing their effects on the model behavior

- For stochastic models statistical analysis of the results is necessary!
  - This means running the models several times for the same parameters, gathering the data and analyzing them in Excell, R or other statistical package.
  - The amount of gathered data can be quite large

Agent-based models (ABM)

Agent

The term "agent" means an active, autonomous and situated unit.

- Autonomous
  - Agents are not directed by an external central unit
- Situated
  - Agents interact in and with some kind of shared environment (for example movement is a type of interaction with environment)
  - Agents interact locally (in space or in a network)

- **Reactive agents** could be relatively simple (without ability to learn or sometimes even without memory) but they don't need to be homogenous
- **Deliberative (or Intelligent) agents** have memory and can have a rich symbolical representation of the enivronment, they can use classical metohods of artificial intelligence (machine learning, neuronal networks, genetic algorithms) to make complex decisions, they can adapt on the changing environment and actions of other agents

Relation between ABM and Multi-agent systems (MAS)

ABM are a subclass of Multi-agent systems (MAS). Typically in MAS agents could be biological or artificial entitities situated in a real world like a group of animals, group of cooperating robots or virtual entities situated in a non-simulated environment like software agents acting in a computer network. In ABM agents are typically software objects (inter) acting in a simulated environment. ABM could be interpreted as models of real-world MAS.

Characteristics of ABM:

- Bottom-up approach (From basic parts to complex interactions - the macro parameters are result of intaractions on the micro level.)
- Time is discrete.
- Basic building blocks are represented by agents (individuals).
  - Agents are defined by their parameters and recurrent functions which define the behaviour of the agents.
  - The behavior is essentially the change of parameters in every step of the model.
  - This change depends on the values of the parameters of the agent, the parameters of other agents and on the local and global parameters of the environment.
- Agents could be adaptive - they change their behaviour in response to the environmental change.
- In some models agents could die and new agents could be introduced.

When to use ABM:

- Complex, non-linear or discrete behavior and interaction of agents
- Non-homogenous and boundedly rational population of agents
- Interaction is local and dependent on some spatial or social structure

Which features of real complex systems can we better understand with the help of ABM?

- Self-organized behavior and decentralized management
- Robustnes and phase transition

Logic and modeling

In every model there are present aspects of **deduction, induction** and **abduction**. But according to the questions we ask the emphasis could be on different types of logical reasoning.

- **Deduction** – we know the principles and try to predict the system behavior.
  - Virus (How will a disease spread through the society?)
  - Traffic (When will the traffic jam arise?)
- **Induction** - we know the global behavior and search for underlying fundamental principles of system dynamics. Is the model robust? Does the model lead to the same or similar behavior for a large range parameter values?
  - Model of predator-prey interactions at NANIA (How does the stability arise in a predator-prey ecosystem?)
  - Termites and Ants (you can also try this alternative ant model) (How does an emergent property arise in biological system?)
  - Flocking (How does the cohesion of the flock arise?)
  - Ethnocentrism (How does a particular kind of behavior evolve in the population?)
  - Segregation (How does the ethnical segregation arise?)
- **Abduction** – we search for the best explanation (basic assumptions and parameters) of specific interesting results.
  - Cooperation (When will the cooperative behaviour become advantageous?)
  - Fire (When will the fire consume a major part of the forrest)

Example: Evolution of cooperation in iterated prisoners dilemma models

Prisoner's dilemma in game theory
Two agents decide between cooperation and non-cooperation and are rewarded after their decisions.

Three basic situations could arise:

- One of the agents cooperates and the second does not. In this case the non-cooperating gets the highest reward A and the cooperating agent the lowest reward D.
- If both agents cooperate they both get the reward B.
- If they both don't cooperate they both get reward C.
- The rewards must satisfy the inequality: A > B > C > D.

Iterated prisonner's dilemma (IPD)
The decisions of agents are repeated and rewards accumulated.

The tournament of different playing strategies surprisingly showed that the best strategy for IPD is very simple:
Tit for Tat: If you cooperate I also cooperate, if you don't cooperate I also don't cooperate.

If we encode the strategies into a simple genome and evolve competing populations of these strategies the Tit for Tat strategy will evolve spontaneously and became dominant for a reasonable range of rewards.

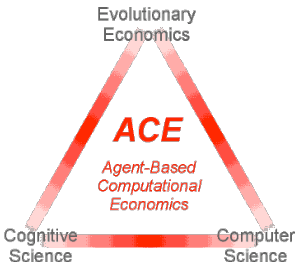
Other examples:

- The evolution of cooperative behavior in spatial environment, see Netlogo model
- The evolution of ethnocentrism - spontaneous evolution of cooperation in ethnic groups with minimal cognitive abilities (without memory), see NetLogo model

Applications:

- Conflict resolutions (initially Axelrod tried to use IPD as a model for political decisions during Cold War)
- Economy

Agent-based computational economics (ACE)



(Graphic by T. Eymann)

Aims of ACE (according to Tesfatsion):

- Empirical understanding (why some macro patterns emerge in economy)
- Normative understanding (how to propose socially desirable economical designs)
- Qualitative understanding (what possible behaviors for what possible parameters can we expect)
- Methodological advancement (to provide methods and tools needed to undertake theoretical studies of economic systems through systematic computational experiments)

ABM and micro-economical models share the bottom-up approach but in other aspects they substantially differ.

> *In the process of* ***formalizing a theory into mathematics*** *it is often the case that one or more — usually many! —* ***assumptions are made for purposes of simplification****; representative agents are introduced, or a single price vector is assumed to obtain in the entire economy, or preferences are considered fixed, or the payoff structure is exactly symmetrical, or common knowledge is postulated to exist, and so on.* ***It is rarely desirable to introduce such assumptions, since they are not realistic and their effects on the results are unknown a priori****, but it is expedient to do so. ... it is typically a* ***relatively easy matter to relax such 'heroic' assumptions-of-simplification in agent-based computational models****: agents can be made diverse and heterogeneous prices can emerge, payoffs may be noisy and all information can be local.(Axtel, 2000)*

Micro-economical models	ABM
analytical solutions	computational synthesis
looking for equilibrium	dynamical systems often without any equilibrium
description of behavior	emergent behavior
homogenous agents	non-homogenous agents
based on variables	based on relations

For comprehensive overview see [ACE web pages by Leigh Tesfatsion].

Two particularly interesting models implemented in NetLogo:

[Artificial stock market] by Carlos Goncalves

[Model of market without intermediation] by Michal Kvasnička

Software tools for ABM

[NetLogo]

Advantages

- Excellent interface
- Easy to learn
- Very easy to implement a model

- Wide support and many well documented and publicly available models
- 3D environment available

Drawbacks

- Implemented in Java - slower than other models
  (2-3 times slower than RePast)
- Not suitable for larger simulations
  (Inability of the older versions (prior to 4.0) to include external code)

Examples of extremely simple NetLogo Forrest fire model and its modification for beginners.

RePast

Advantages

- Relatively fast (it is basically a set of Java libraries)
- Wide support
- Suitable for larger simulations

Drawbacks

- Relatively hard to learn
- Longer implementation times
  (It usually takes 2-3 times more effort to implement the same simple model in RePast then in NetLogo)

For other ABM tools and more detailed comparation see http://en.wikipedia.org/wiki/Comparison_of_agent-based_modeling_software

Resources about ABM

A guide for newcomers to agent based modeling in the social sciences - by Robert Axelrod and Leigh Tesfatsion

ACE web pages by Leigh Tesfatsion

Agent-based modeling: Methods and techniques for simulating human systems by Eric Bonabeau

Seeing around corners - a popular article about ABM by Jonathan Rauch

Why agents? On the varied motivations for agent computing in the social sciences - an elaborated analysis of relations between ABM and classical analytical models, by Robert Axtell

From factors to actors: Computational sociology and agent-based modeling - sociological approach to ABM by Michael W. Macy and Robert Willer

Complexity of Cooperation Web - by Robert Axelrod

Twenty Years on: The Evolution of Cooperation Revisited - an overview by Robert Hoffmann

Parochial altruism resources

References

Albert, R. *J Cell Sci* 2005;118:4947-4957

Axelrod, R. (2003): Advancing the Art of Simulation in the Social Sciences. *Japanese Journal for Management Information System*, Special Issue on Agent-Based Modeling, Vol. 12, No. 3, Dec. 2003.

Axelrod, R. (1987): The Evolution of Strategies in the Iterated Prisoner's Dilemma. In Lawrence Davis (ed.),*Genetic Algorithms and Simulated Annealing*. London: Pitman, and Los Altos, pp. 32-41. available on WWW: < http://www-personal.umich.edu/~axe/research/Evolving.pdf>.

Barabasi, A. L. Bonabeau, E. *Scientific American*. New York: May 2003. Vol. 288, Iss. 5; p. 60.

Cools, S.B., C. Gershenson, and B. D'Hooghe (2006): *Self-organizing traffic lights: A realistic simulation*. available on WWW: < http://uk.arxiv.org/abs/nlin.AO/0610040>.

Corning, Peter A. (2002), The Re-Emergence of "Emergence: A Venerable Concept in Search of a Theory. *Complexity* 7(6): 18-30

De Wolf, Tom & Tom Holvoet (2005), "Emergence Versus Self-Organisation: Different Concepts but Promising When Combined", Engineering Self Organising Systems: Methodologies and Applications, Lecture Notes in Computer Science: 3464, 1-15. Available on WWW:< http://www.cs.kuleuven.be/~tomdw/publications/pdfs/2005esoa04lncs.pdf >.

Edelman, G. Tononi, G. *Universe of consciousness*. Basic Book. New York. 2000.

Gell-Mann, M. What Is Complexity?. *Complexity*. 1995, Vol 1, no.1.

Huang, Chung-Yuan, Sun, Chuen-Tsai and Lin, Hsun-Cheng (2005). Influence of Local Information on Social Simulations in Small-World Network Models. *Journal of Artificial Societies and Social Simulation* 8(4)8 http://jasss.soc.surrey.ac.uk/8/4/8.html.

J. Goldstein. Emergence as a Construct: History and Issues. *Emergence* 11: 1999, pp. 49-72.

Hammond, R.A., Axelrod, R. (2005): The Evolution of Ethnocentrism. *The Journal of Conflict Resolution*. Beverly Hills: Dec 2006. Vol. 50, Iss. 6; pg. 926, 11 pgs.

Hoffmann, R. (2000): Twenty Years on: The Evolution of Cooperation Revisited. *Journal of Artificial Societies and Social Simulation* vol. 3, no. 2. Available on WWW:< http://jasss.soc.surrey.ac.uk/3/2/forum/1.html>.

Kauffman, S. (1995): *At Home in the Universe: The Search for the Laws of Self-Organization and Complexit*y. Oxford University Press.

Langton, C. G. Computation at the edge of chaos. *Physica* D, 42, 1990.

Lovelock, J.E.; Margulis, L. (1974). "Atmospheric homeostasis by and for the biosphere- The Gaia hypothesis". *Tellus* 26 (1): 2–10.

Macy, W.M., Willer, R. (2002): From factors to actors: Computational sociology and agent-based modeling. Annual *Review of Sociology*; 2002; 28, pg. 143

61. Paul Rendell (2005). [online] . A Turing Machine in Conway's Game of Life. [cit. 1.5.2010]. Dostupné na WWW: < http://rendell-attic.org/gol/tm.htm >.

Sporns, O. Tononi, G. and Edelman, G.M. Connectivity and complexity: the relationship between neuroanatomy and brain dynamics. *Neural Networks*, 13:909–992, 2000. available on WWW <http://php.indiana.edu/~osporns/nn connectivity.pdf>.

Sporns, O. Tononi, G. Classes of network connectivity and dynamics. *Complexity*, 7:28–38, 2002. available on WWW <http://php.indiana.edu/~osporns/complexity 2002.pdf>.

Shalizi, C. R. and Crutchfield, J. P. Computational mechanics: Pattern and prediction, structure and simplicity. *Journal of Statistical Physics*, 104:817–879, 2001. available on WWW: <http://arxiv.org/abs/cond-mat/9907176>.

Shalizi, C. R. Optimal nonlinear prediction of random fields on networks. *Discrete*

Mathematics and Theoretical Computer Science, AB(DMCS):11–30, 2003. available on WWW: <http://arxiv.org/abs/math.PR/0305160>.

Shalizi, C. R. Shalizi, L. K. and Haslinger, R. Quantifying self-organization with optimal predictors. *Physical Review Letters*, 93:118701, 2004. available on WWW: <http://arxiv.org/abs/nlin.AO/0409024>.

Wuensche, A. Discrete Dynamical Networks and their Attractor Basins. In: *Complex Systems '98*. University of New South Wales, Sydney, Australia. 1998.

Wolfram, S. *A New Kind of Science*. Wolfram Media. 2002.

Young, L. The organization of neural system in the cerebral cortex. *Proceedings of the Royal society*. 1993, Series B, 252: 13-18.

Literature in Czech:

Brockman, J. (ed.). *Třetí kultura*. Academia. Praha. 2008.

Burian, J. Reaktivní multiagentní modely v ekonomii. Doktorská práce. VŠE Praha. 2010. Dostupné na WWW:< http://eldar.cz/cognition/complex/articles/burian_rACE.pdf >.

Burian, J. Multiagentní model transakčních nákladů na finančních trzích. In: Kvasnička, V. Kelemen, J. Pospíchal, J.(eds.). *Kognice a umělý život VIII*. 2008. Slezská univerzita. Opava. Dostupné na WWW:< http://eldar.cz/honza/articles/burian_ace_fin_market_tc.doc >.

Burian, J. Evoluce averze k riziku a nejistotě. In: Kvasnička, V. Kelemen, J. Pospíchal, J.(eds.). *Kognice a umělý život VII*. 2007. Slezská univerzita. Opava. Dostupné na WWW:< http://eldar.cz/honza/articles/burian_evoluce_averze.doc >.

Burian, J. Samoorganizace a kognice. In Kelemen, J., Kvasnička V. (eds.). *Kognice a umělý život VI*. Opava. Slezská univerzita v Opavě. 2005. Dostupné na WWW:< http://eldar.cz/honza/articles/burian_samoorganizae_kognice_final.doc>.

Capra, F. *Tkáň života*. Academia. Praha. 2004.

Havel, I.M. Přirozené a umělé myšlení jako filosofický problém In: *Umělá inteligence III*. (V. Mařík a kol., editoři), Academia, Praha 2001, s. 17-75. Dostupné na WWW: < http://www.cts.cuni.cz/reports/1999/CTS-99-11.htm >.

Kauffman, S. *Čtvrtý zákon*. Paseka. Praha a Litomyšl. 2004.

Kubík, A. *Inteligentní agenty*. Grada. 2004.

Mandelbrot, B. *Fraktály*. Mladá fronta. Praha. 2003.

Mařík, V. Štěpánková, O. (Eds.) Umělá inteligence III (kapitola Umělý život). Academia. Praha. 2003.

Salamon, T. *Design of Agent-Based Models : Developing Computer Simulations for a Better Understanding of Social Processes.* Repin, Czech Republic: Bruckner Publishing. 2011. WWW: <http://www.designofagentbasedmodels.info>

Highfield, R. Coveney, P. *Mezi chaosem a řádem*. Mladá Fronta. Praha. 2003.