15-382 Collective Intelligence – S19

Lecture 1:
Introduction

Teacher:
Gianni A. Di Caro

جامعة كارنيجي ميلون في قطر
Carnegie Mellon University Qatar

Collective Intelligence?

Group of ants carrying a big prey (lizard) to colony nest

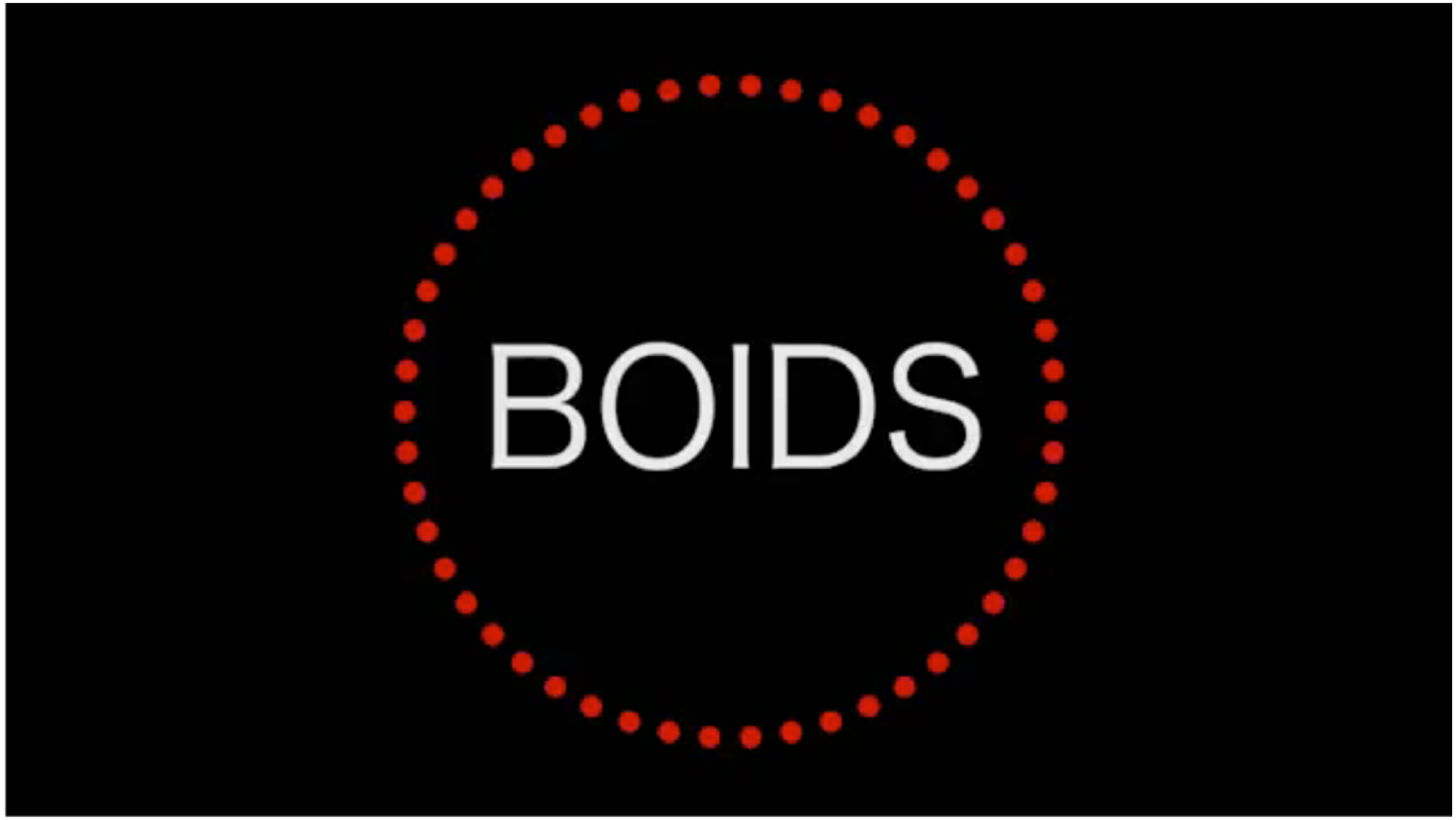
Collective Intelligence?

Ants fill a gap by growing from both sides an ant bridge

Collective Intelligence?

Bird flocking, smooth, mesmerizing

COLLECTIVE INTELLIGENCE?



Boids flocking! (from C. Reynolds' work, 1986)

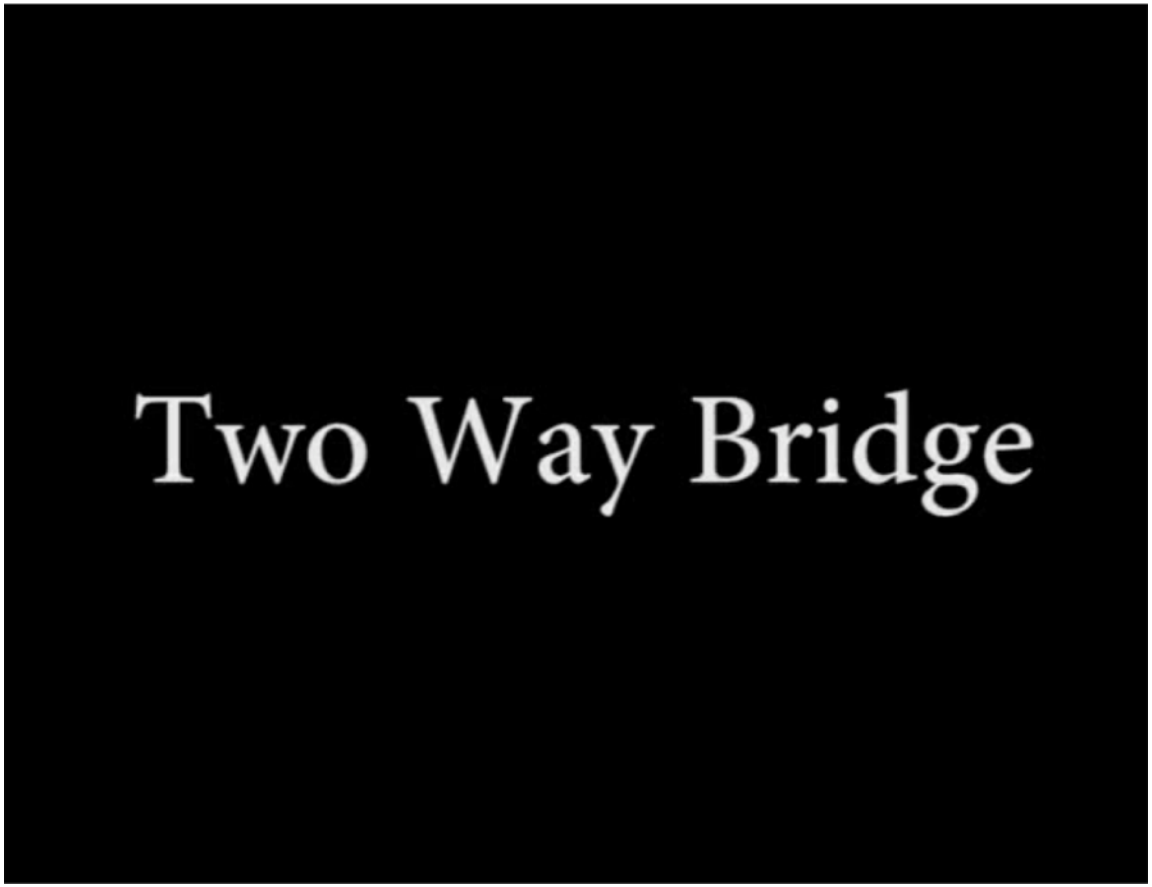
Collective Intelligence?

Fish schooling (sardines)

Collective Intelligence?

Fireflies synchronized flashing (in Thailand)

Collective Intelligence?



Two-ways and three-ways bridge (homemade) settings for ant colonies: pheromone trails allow the colony finding the shortest way

Collective Intelligence?

Ant colonies are able to discover short-cuts!

Collective Intelligence?

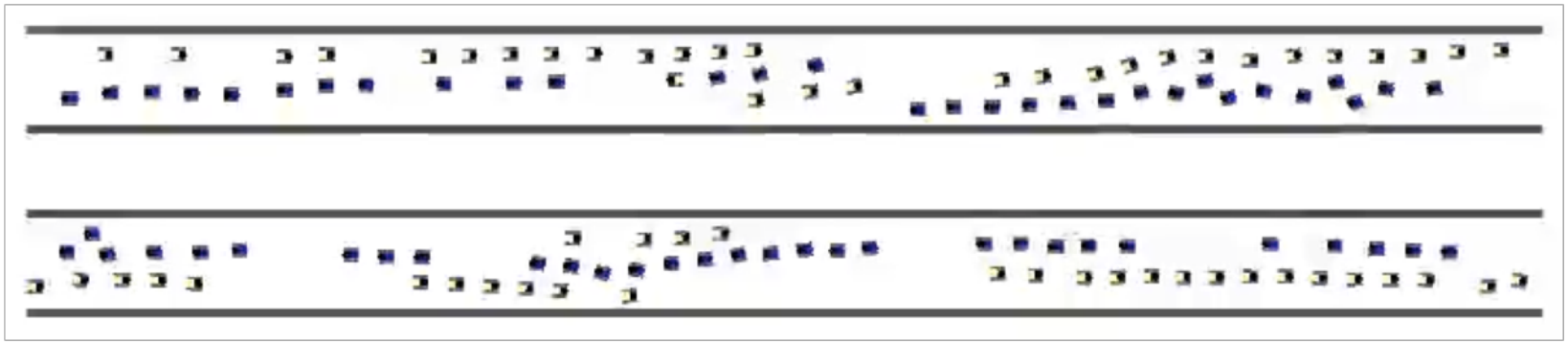
Swarms of locusts: flocking, leader follower behaviors

Collective Intelligence?

Swarm of robots dynamically self-organizing

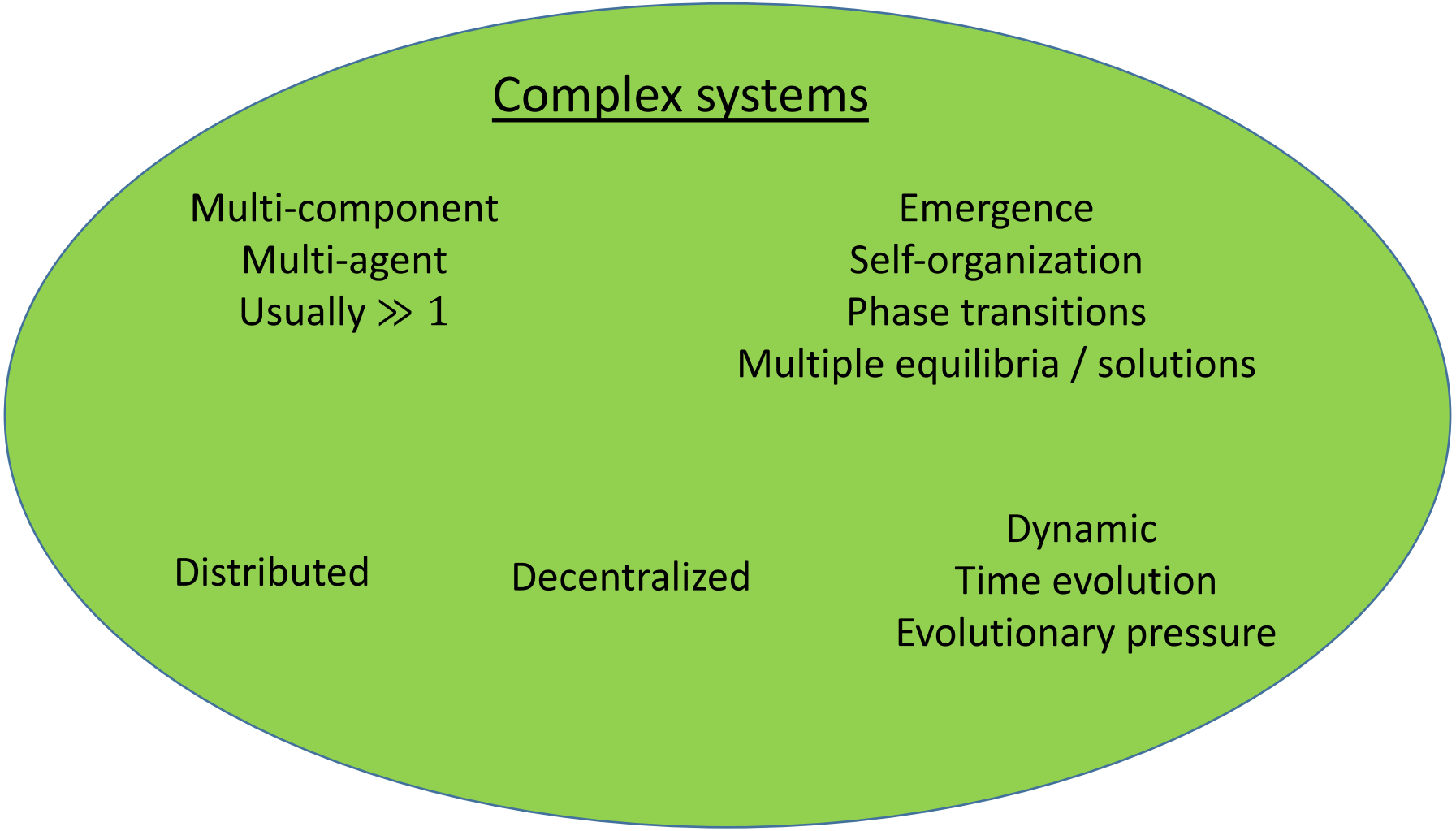
COLLECTIVE INTELLIGENCE?

Shibuya crossing



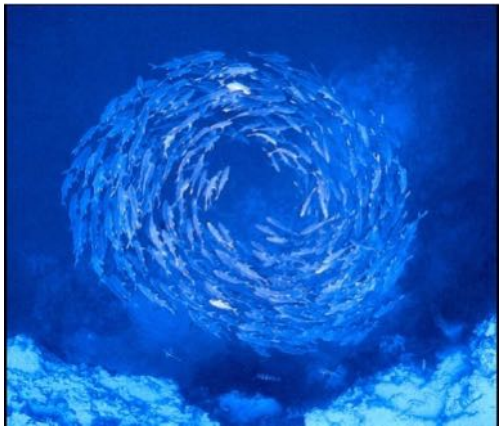
Pedestrians and mobile robots creating ordered flows

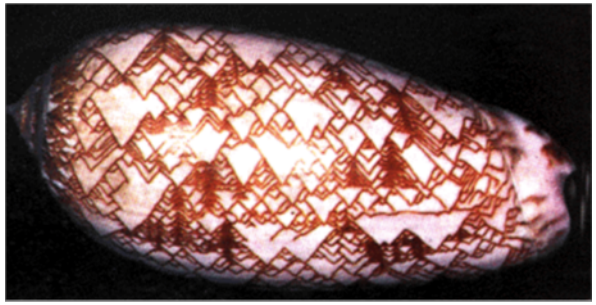
COLLECTIVE *INTELLIGENCE*?



Modeling? **Properties?** **Control?**

Complex ≠ Complicated





EMERGENCE

"Precursors" of the notion of *emergence:*

- Whole before its parts
- *Gestalt:* a configuration or pattern of elements so unified as a whole that it cannot be described merely as a sum of its parts
- *Non-linearity*

$$f(a+b) \neq f(a) + f(b)$$

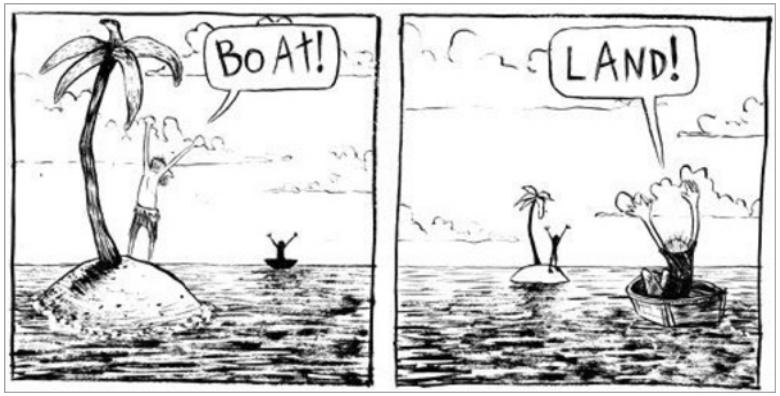
Emergence:

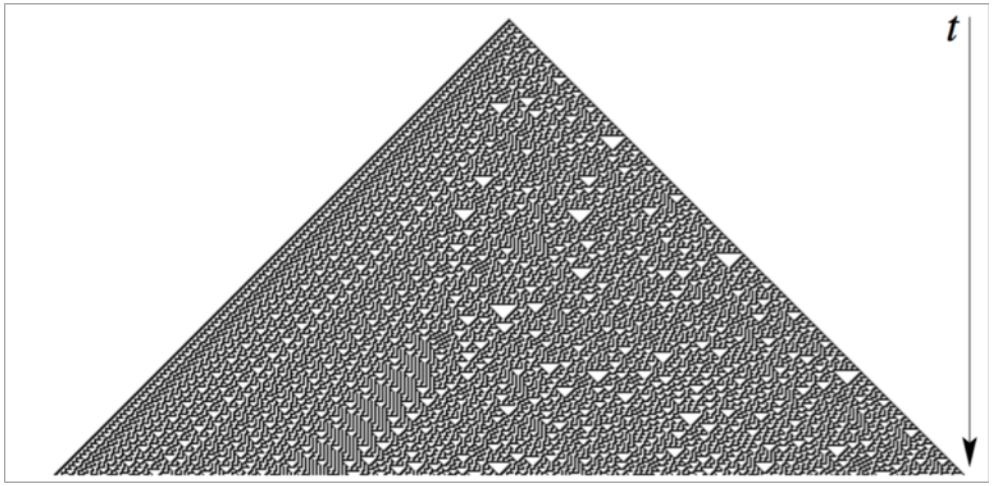
A system exhibits emergence when there are coherent *emergents* at the macro-level that <u>*dynamically*</u> arise from the (*localized*) interactions between the parts at the micro-level.

Such emergents are <u>novel w.r.t. the individual parts of the system</u>.

EMERGENCE: SPACE-TIME SCALES

- What are the **micro** and **macro** *levels?*
- → The ***Observer*** matters: space-time scales

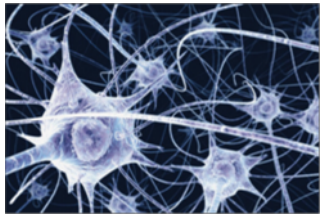




EMERGENCE ~ RADICAL NOVELTY

- **Radical novelty:** The collective behavior is not readily understood from the behavior of the parts. The collective behavior is, however, *implicitly contained* in the behavior of the parts if they are studied in the context in which they are found.
- Emergent properties cannot be studied by physically taking a system apart and looking at the parts (=*reductionism*). They can, however, be studied by looking at each of the parts in the context of the system as a whole.
- **<u>Problem:</u>** Predictability of the behavior / evolution of the system!?

→
?



→
?

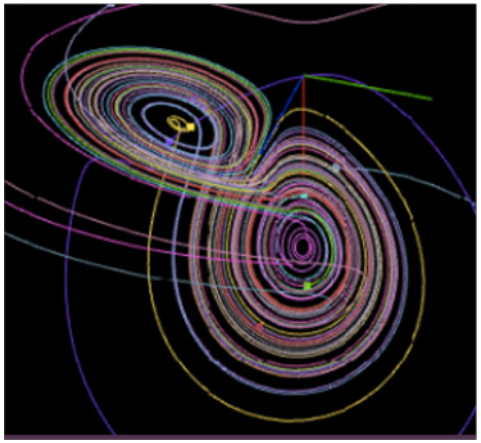
EMERGENCE: DECENTRALIZED CONTROL

- Only *local mechanisms* to influence the global behavior.
- There is no central control, i.e. no single part of the system directs the macro-level behavior: The actions of the parts are controllable. The whole is not directly controllable → No leader!
- This characteristic is a direct consequence of the radical novelty that is required for emergence. Centralized control is only possible if that central part of the system has a representation of the global behavior (e.g. a plan)
- Not even centralized instruction sets with a representation of the global behavior (e.g., set provided by an 'orchestra' director)

SELF-ORGANIZATION

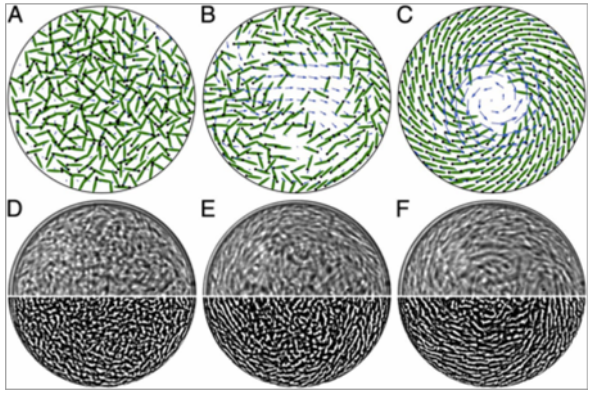
Self-organization is a dynamical and adaptive process where systems acquire and maintain *structure* themselves, without external control.

State-space reduction



(strange attractor)

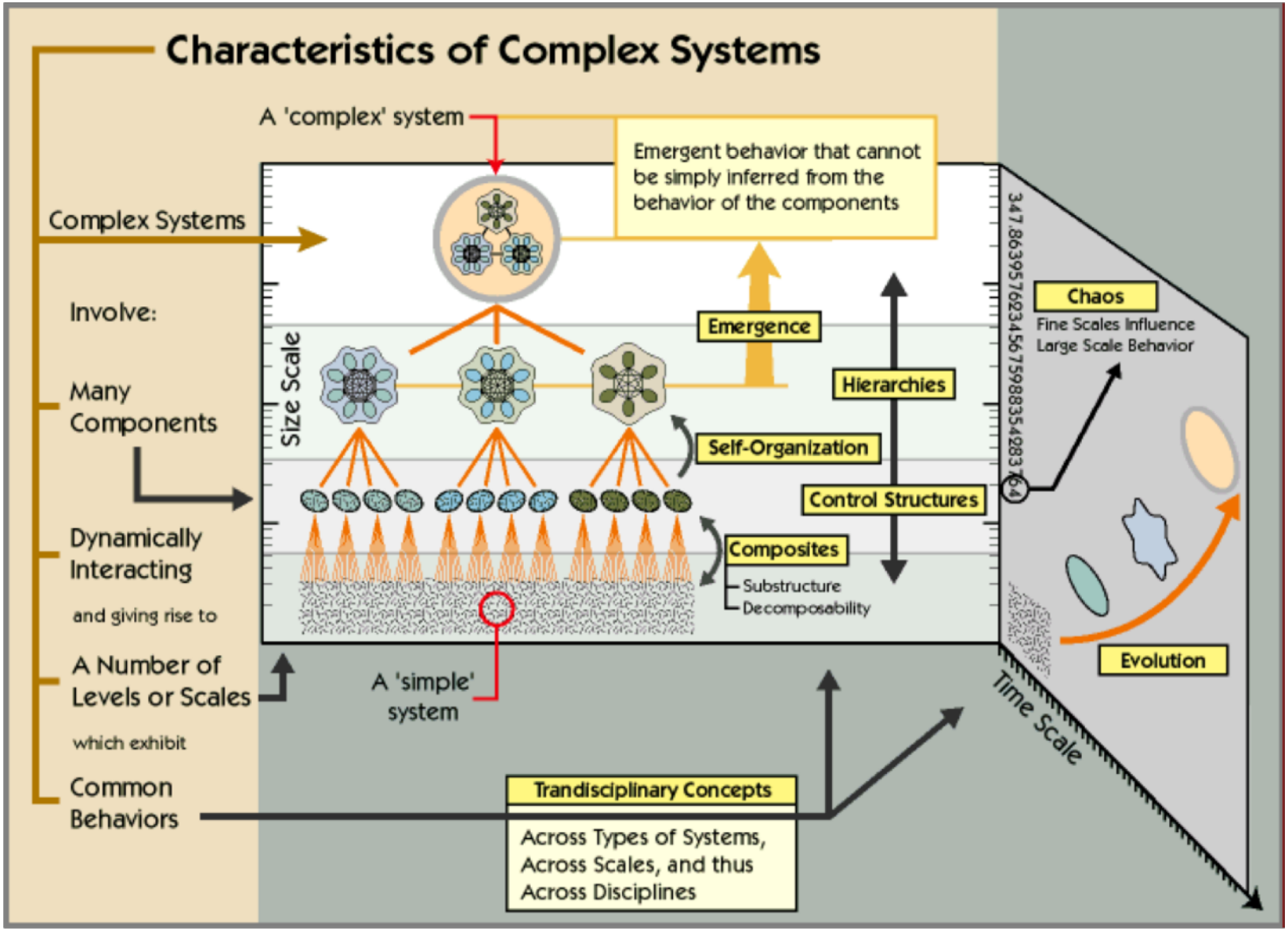
The *structure* can be spatial, temporal or functional



Complex Systems: Fingerprints

- Multi-agent / Multi-component
- Decentralized: neither central controller, nor representation of global patterns/goals
- Possibly (not necessarily) with a large number of components
- Localized interactions (allowing propagation of information)
- Emerging and / or Self-Organizing properties
- Agents do not need to be “complex”
- Dynamic: Time and space evolution of the system

A view of Complex Systems



COMPLEX SYSTEMS: GOALS

- **Goals:**
  - Understanding and modeling natural systems for prediction and control
  - Design artificial systems that enjoy complex systems *properties*

- **Properties** (that *could* be obtained):
  - Robustness
  - Parallelism
  - Adaptivity
  - Fault-tolerance

  - ***A bottom-up way of "computing" ….***

BOTTOM-UP VS. TOP-DOWN

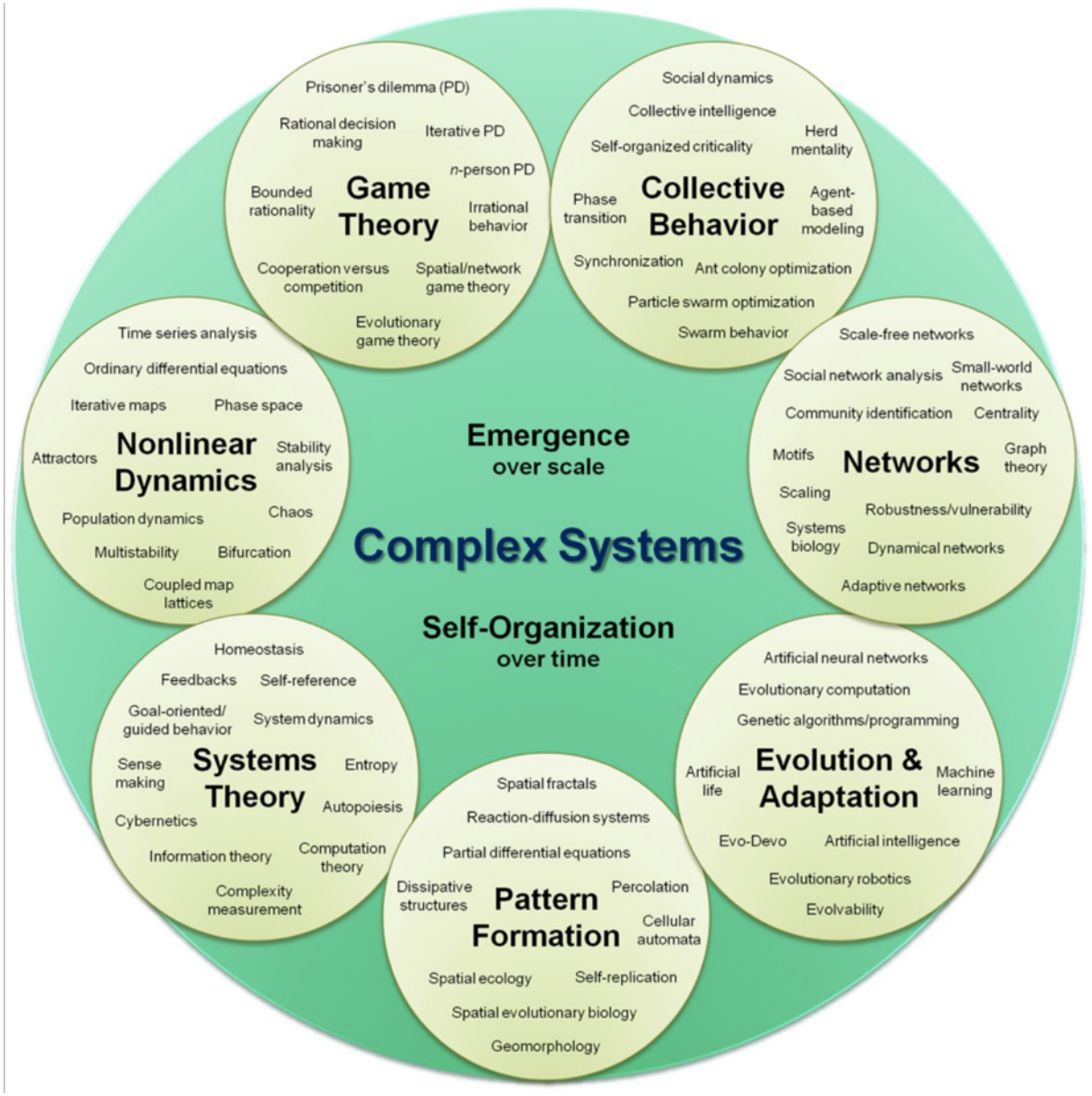
- Ontogenetic and phylogenetic evolution has (necessarily) followed a bottom-up approach (grassroots) to design systems:
  - Instantiation of the basic units (atoms, cells, organs, organisms, individuals, . . .) composing the system and let them (self-)organize to generate more complex/organized system-level behaviors and/or structures
  - Population + Interaction protocols are "more important" than the single modules
- On the other hand, from an engineering point of view we can also choose a top-down approach:
  - Acquisition of comprehensive knowledge about the problem/system to deal with, analysis, decomposition, definition of a possibly optimal strategy

- **Bottom-up programming:** instantiate the single (possibly relatively simple) components, their interaction protocols and topology, local information exchange, (hope) self-organization …→ *Output:* (Useful) Emerging patterns and functionalities (if any), likely not "precise" but robust, scalable, adaptive
- **Top-down programming** ("regular programming"): write <u>precise instructions</u> for the task to be executed in a variant of classical VonNeumann architecture

BOTTOM-UP CHALLENGES

- Given a task/problem to deal with, a number of design choices:
  1. Characteristics/skills of the agents
  2. Size of the population (related to the choice 1.)
  3. Neighborhood definition
  4. Interaction protocols and information to exchange
  5. Where the information is updated (agent, channel, environment)
  6. Use or not of randomness (or, heuristic decisions)
  7. Synchronous or asynchronous activities and interactions
  8. ...
- Lots of parameters
- Predictability and efficiency are important issues
- Is a top-down approach better?

General Map of Complex Systems' topics



Our set of collective intelligence topics

Week	Topics
1	**Introduction:** Complexity, networks, self-organization, emergence, multi-agents, swarms
	Dynamical systems: Population models, differential equations
	Dynamical systems: Stability, bifurcations
2	**Dynamical systems:** Discrete-time recurrence equations, logistic model
	Dynamical systems: Deterministic chaos
3	**Cellular automata:** Discrete-time spatial models, properties
	Cellular automata: Pattern formation, applications
4	**Networks:** Topological properties, topology and network computation
	Networks: Information diffusion, gossip protocols, topology and CA
5	**Swarm intelligence:** Dynamic topology, Particle Swarm Optimization (PSO) for optimization
	Swarm intelligence: PSO for navigation and foraging in robot swarms
6	**Swarm intelligence:** Stigmergic communications, Ant-inspired algorithms
	Swarm intelligence: Ant Colony Optimization (ACO)
7	**Task allocation:** Division of labor in animal societies, ant-inspired clustering
	Task allocation: Market-based models

Our set of collective intelligence topics

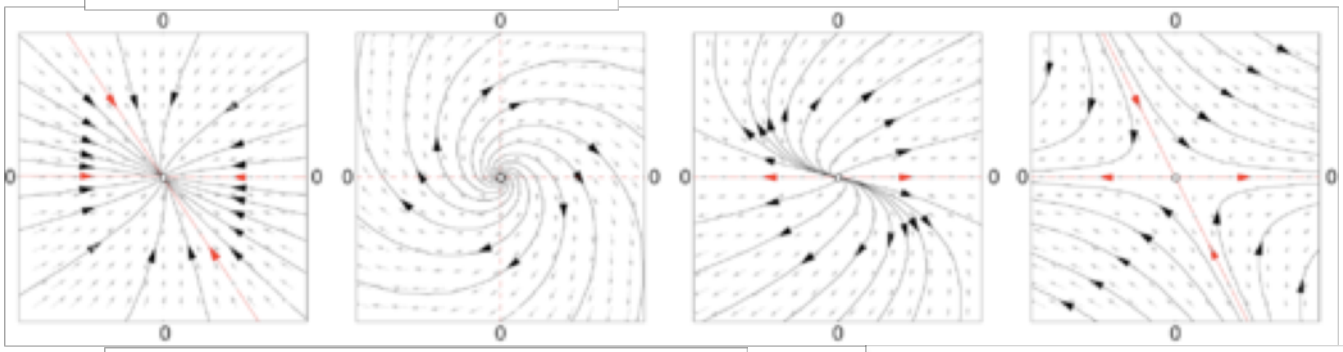
8	**Social decision-making and consensus:** Crowd wisdom, quorum responses, leaders/followers
	Social decision-making and consensus: Distributed consensus
9	*Spring break*
10	**Social decision-making and consensus:** Social choice, theory of voting,
	Social decision-making and consensus: Designing voting rules
11	**Game theory:** Minimax theorem, Equilibria concepts
	Game theory: Correlated equilibria, Leader-follower games
12	**Game theory:** Evolutionary games
	Game theory: Social welfare, Optimality models
13	**Pattern formation:** Moving in a crowd
	Pattern formation: Flocking
14	**Neural models:** Neural networks, perceptrons
	Neural models: Self-organizing maps
15	**Neural models:** Self-organizing maps in deep learning
	Case study, Q&A

DYNAMICAL SYSTEMS

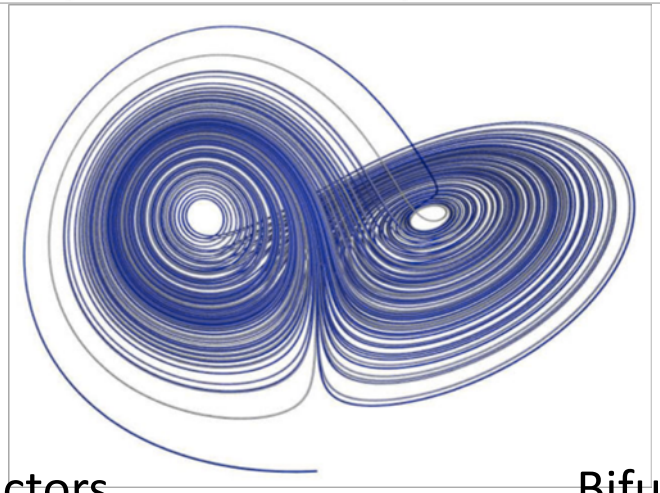
$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = \dot{\mathbf{x}} = \mathbf{X}(\mathbf{x})$$

$$\mathbf{x}_{t+1} = \mathbf{f}(\mathbf{x}_t)$$

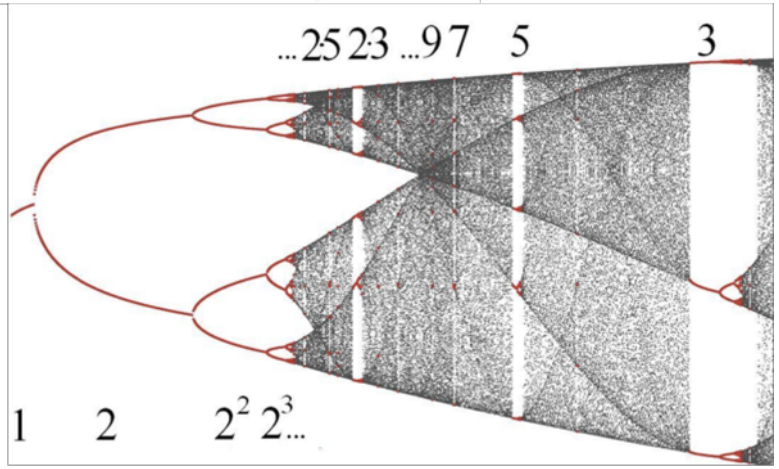
$$\dot{N}(t) = \frac{\mathrm{d}N}{\mathrm{d}t} = rN(t)\left(1 - \frac{N(t-T)}{K}\right)$$



Time evolution (depending on initial conditions)



Attractors



Bifurcations, dependence on parameters

DYNAMICAL SYSTEMS

Basic ingredients:

State variables : $\underline{\boldsymbol{x}} = (x_1, x_2, \ldots, x_n)$

Evolution operator $x_i(t) = \Phi_i(t; x_1(t_0), \ldots, x_n(t_0)) \quad i = 1, \ldots, n \quad t \geq 0$

Initial condition $x_i(t_0)$ $i=1, \ldots n$ + Local evolution laws

<u>In continuous time</u>: $t \in IR$, differential equations

$$\frac{dx_i(t)}{dt} = f_i(x_1(t), x_2(t), \ldots, x_n(t)) \qquad i = 1, \ldots, n$$

<u>In discrete time</u>: $t \in IN$, difference equations (iterated maps)

$$x_i(t+1) = f_i(x_1(t), x_2(t), \ldots, x_n(t)) \quad i = 1, \ldots n$$

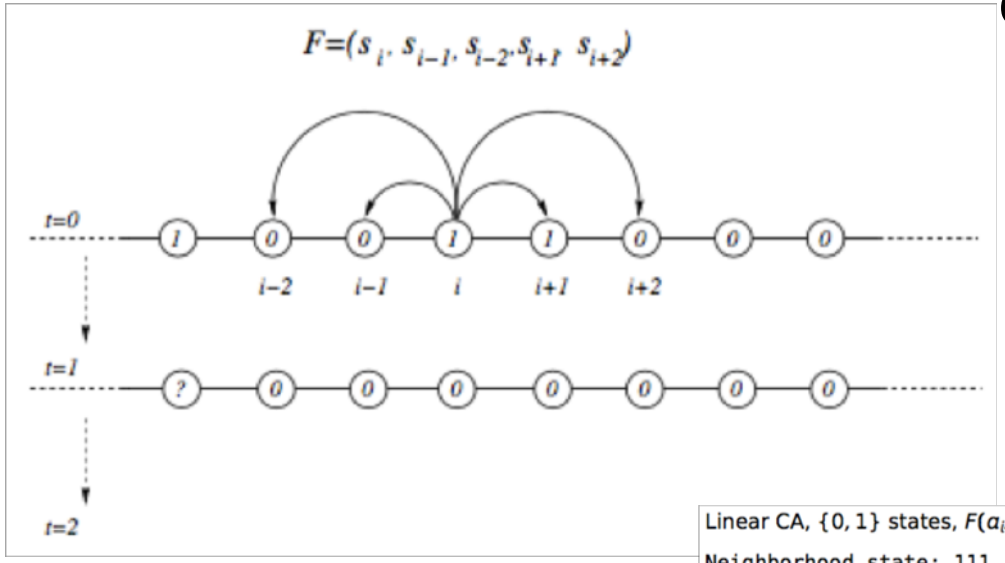
These are sistems of <u>first order</u> <u>autonomous</u> evolution equations.

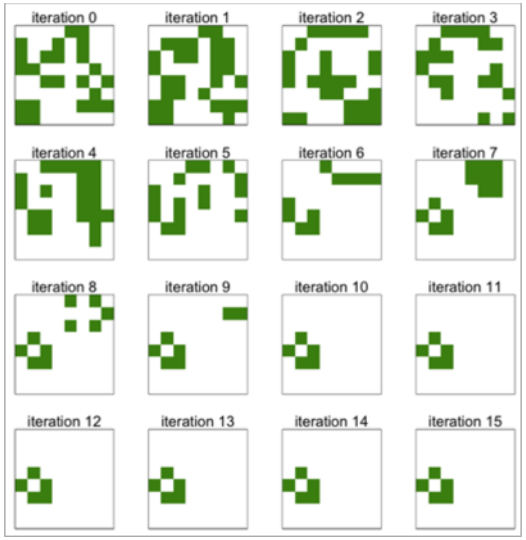
We shall see how higher order equations, as well as nonautonomous equatons can be reduced to this kind of evolution equations

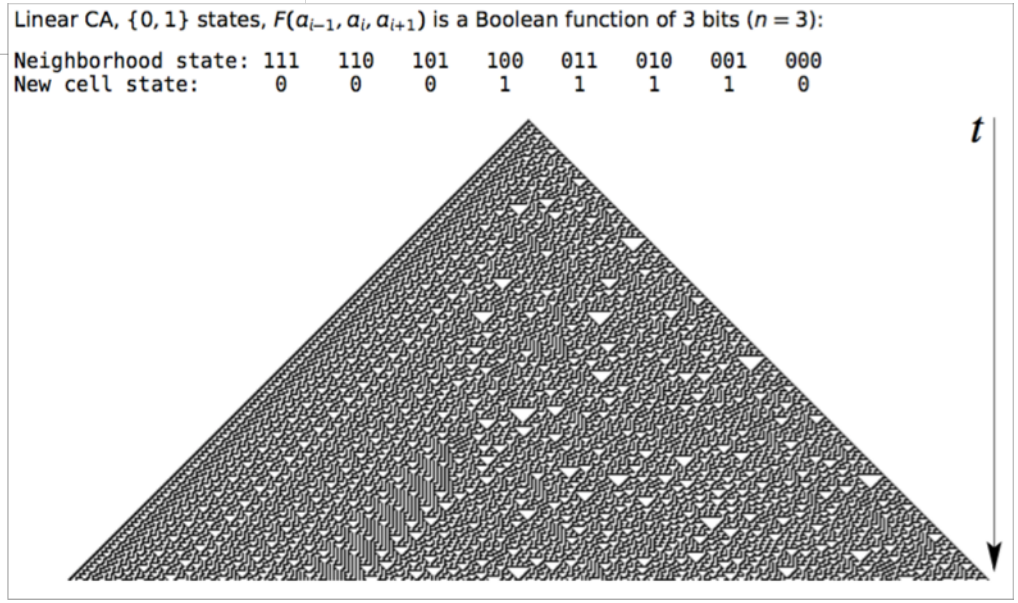
Cellular Automata

Time-discrete spatially-dependent dynamical systems

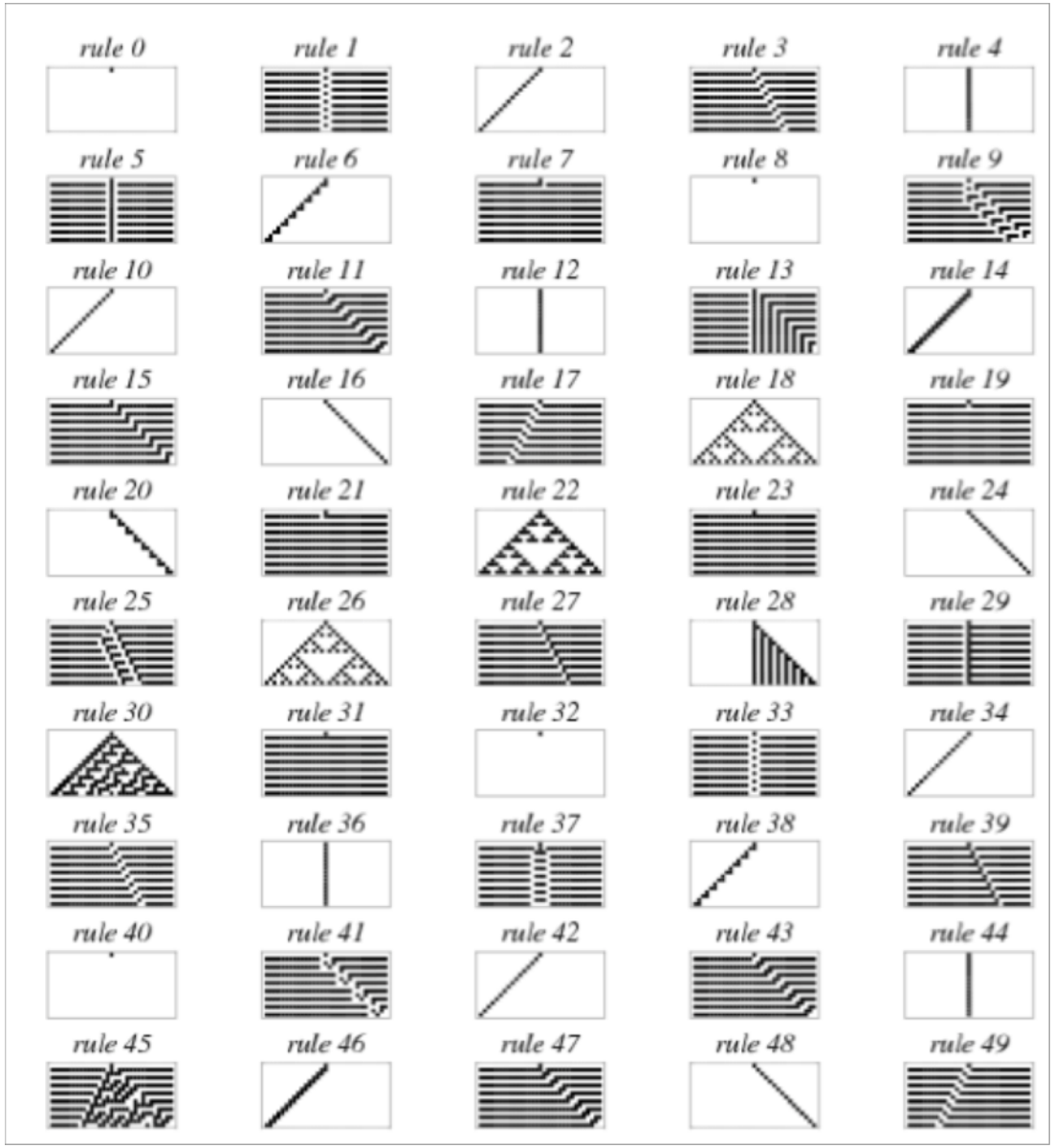
Capable of universal computation





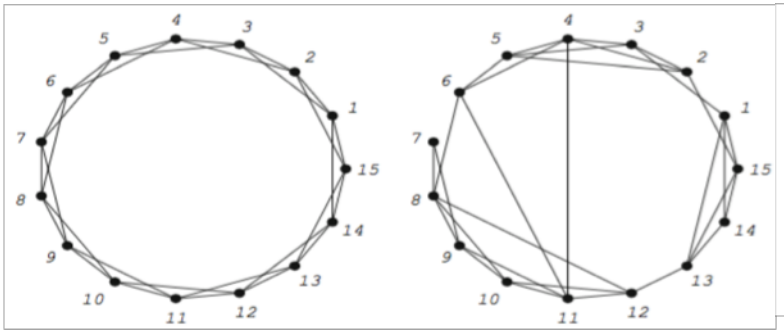


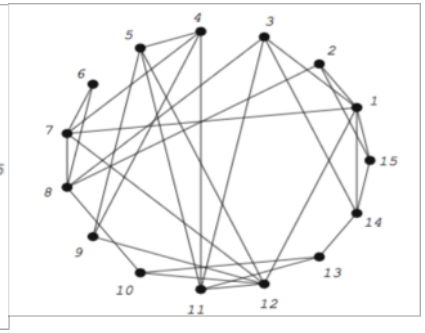
Cellular Automata



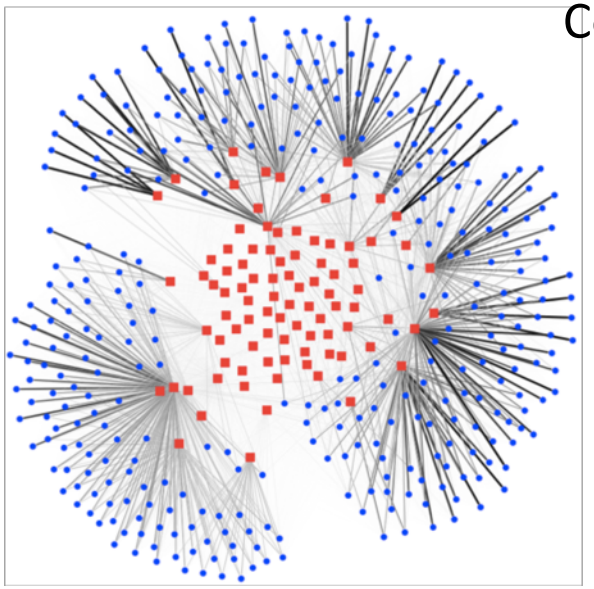
NETWORKS

Connectivity matters: Information dissemination, gossiping, epidemic models

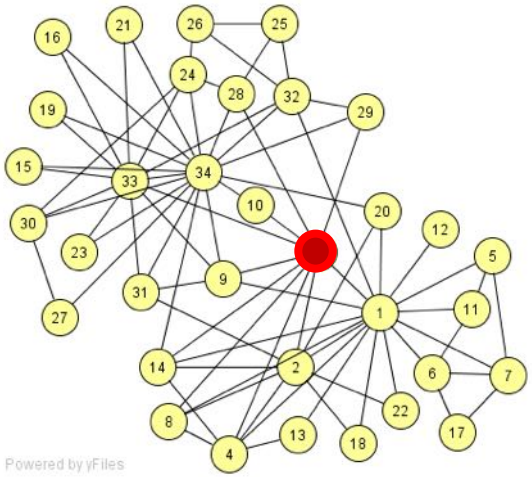


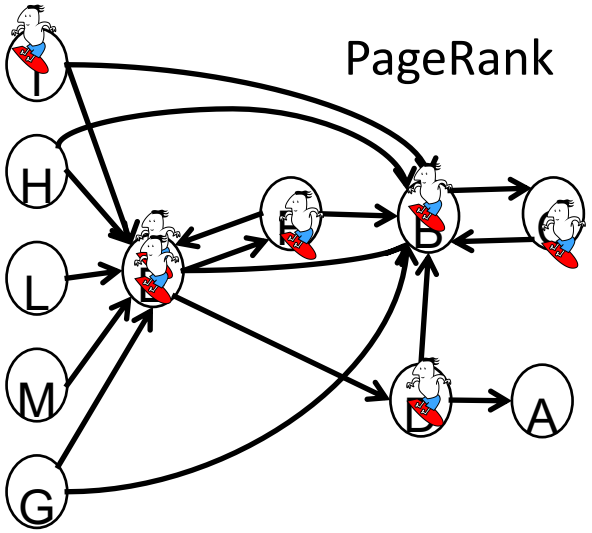


Random, small-world, scale-free topologies

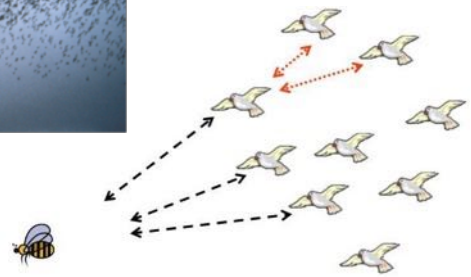


Centrality measures

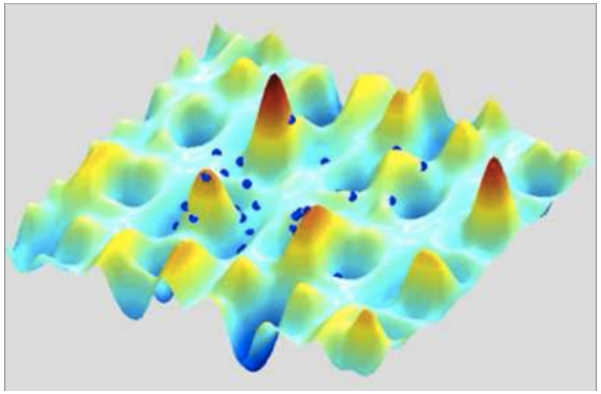




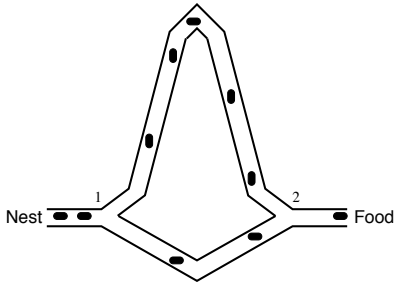
Swarm Intelligence

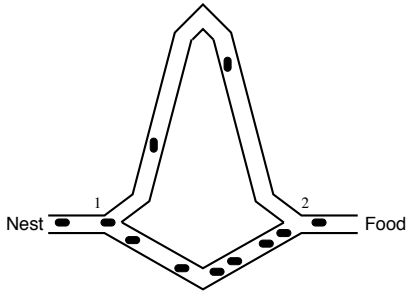


Social neighborhood

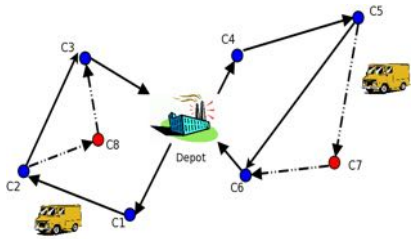


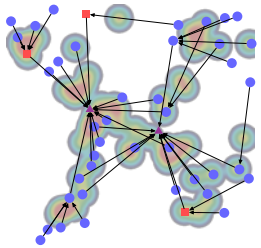
Particle Swarm Optimization (PSO)





Stigmergy

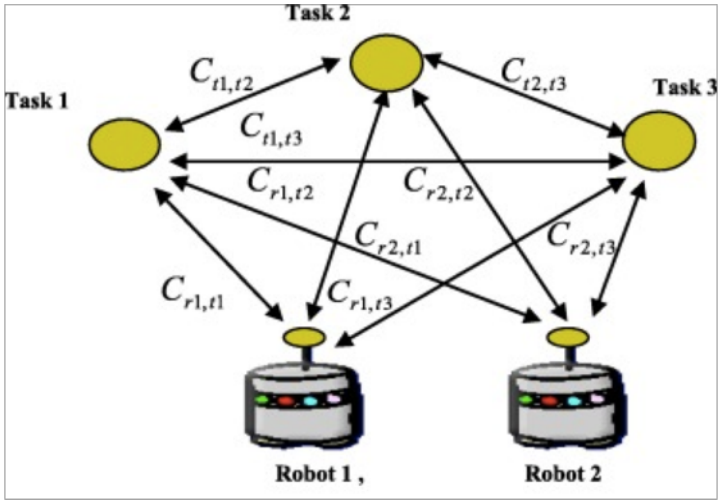


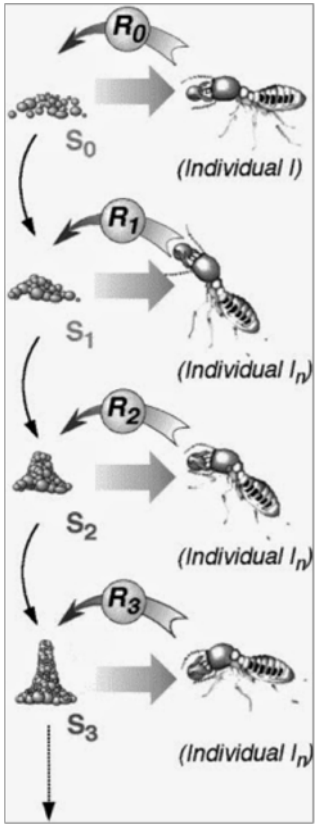


Ant Colony Optimization (ACO)

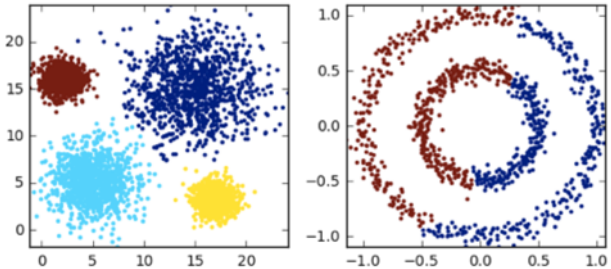
Task / Data Allocation

Who does what…?

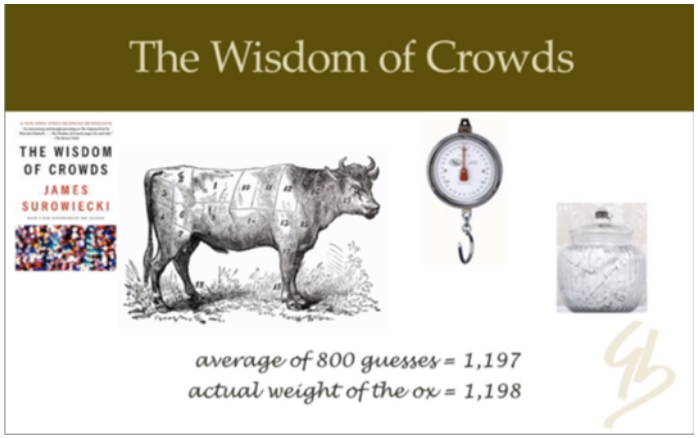




Division of labor
Specialization of work
Automatic task / goods allocation
Data / Object clustering

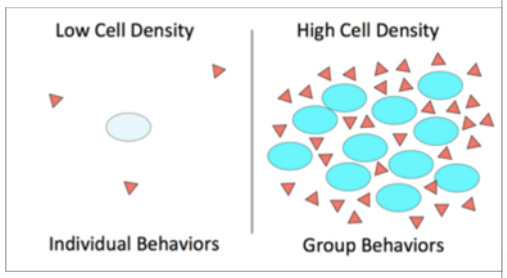


Social Decision-making, Consensus

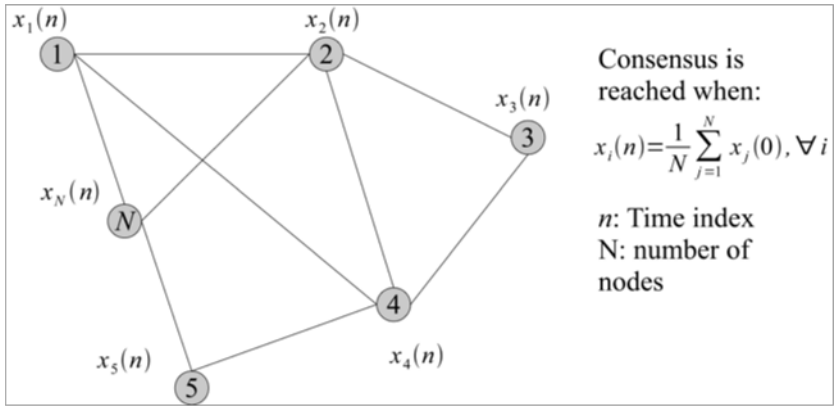


Leaders / Followers

Quorum-based responses

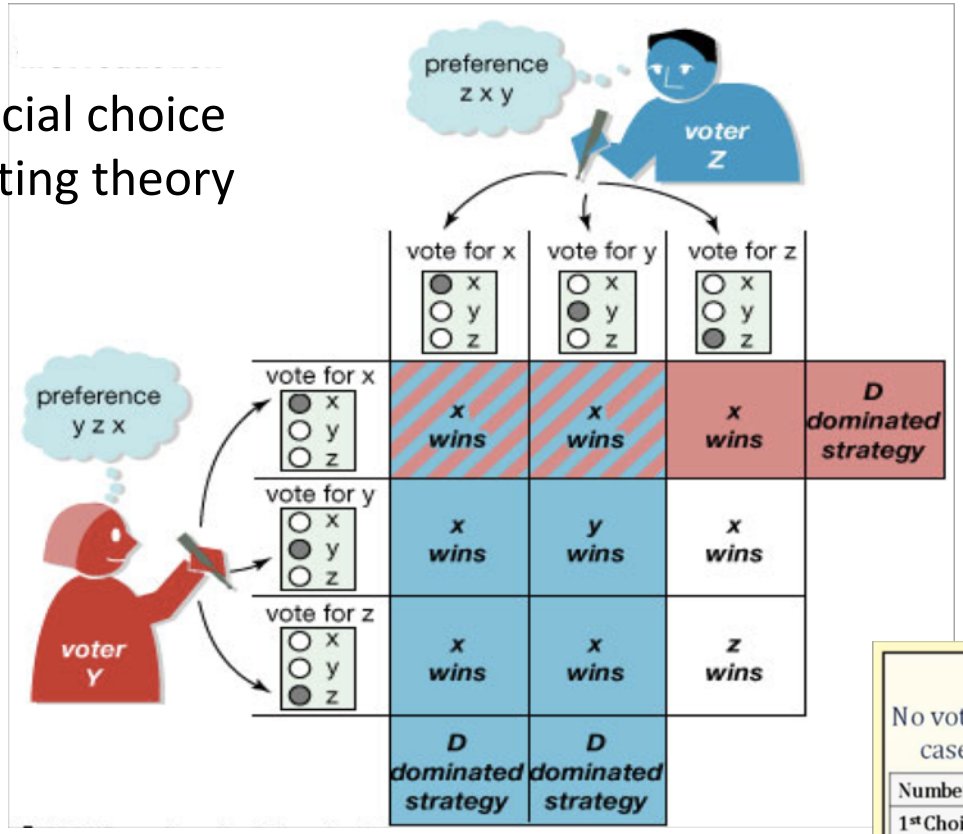


Distributed consensus



Social Decision-making, Consensus

Social choice
Voting theory



Ordinal method of Condorcet (1743–94)

Book *Essai sur l'application…* (1785)
Condorcet paradox: Cyclic majority

Preference	A	B	C
	B	C	A
	C	A	B
	8	7	6

A > B > C > A → A > B > C
14:7 15:6 13:8
weakest link

Arrow's Impossibility Theorem

No voting system can satisfy all four fairness criteria in all cases.

Number of Voters	10	10	10	10
1st Choice	A	B	C	D
2nd Choice	B	C	D	A
3rd Choice	C	D	A	B
4th Choice	D	A	B	C

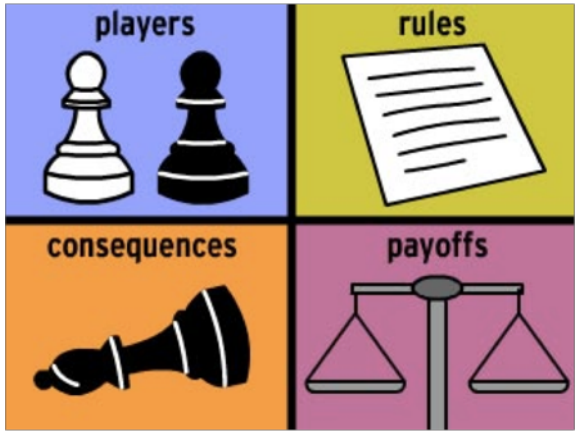
A: 10/40 = 25%
B: 10/40 = 25%
C: 10/40 = 25%
D: 10/40 = 25%

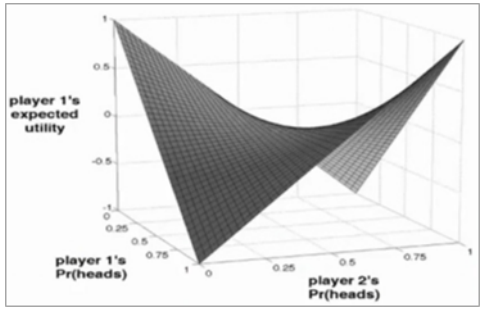
A vs B: 30 to 10
A vs C: 20 to 20
A vs D: 10 to 30
B vs C: 20 to 20
B vs D: 20 to 20
C vs D: 30 to 10

- No candidate has a majority.
- No Condorcet Winner.
- In fact regardless of the candidate, ¾ of the voters prefer someone else.
- This is called Condorcet's Voting Paradox.

GAME THEORY

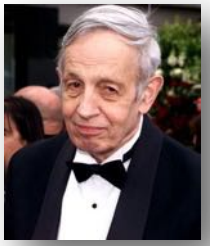
Conflict (and cooperation) in multi-agent systems
Equilibrium concepts: minimax, Nash, correlated, leader-follower



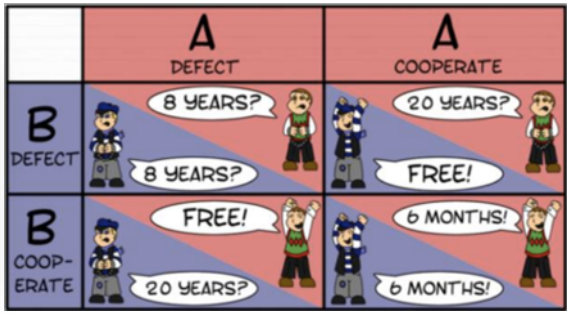


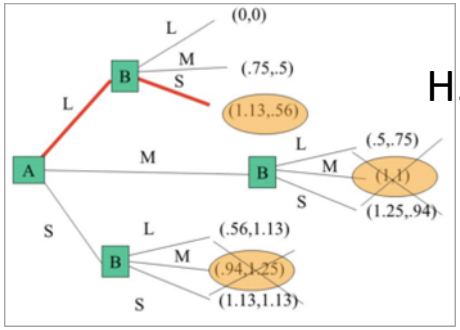
J. von Neumann
(a pure genius)

R. Aumann
(2005 Nobel)



J. Nash
(1994 Nobel)

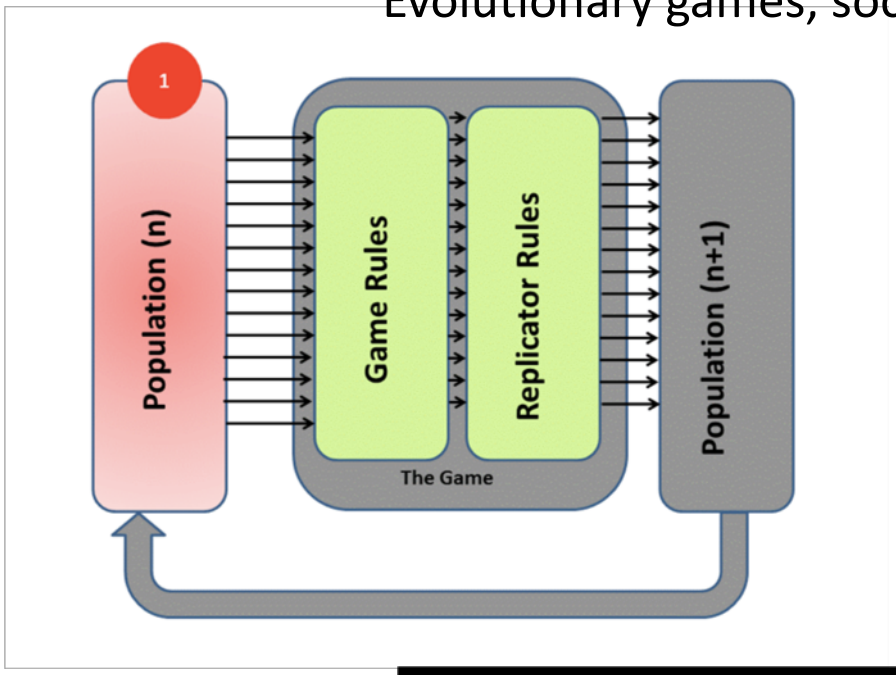




H. von Stackelberg

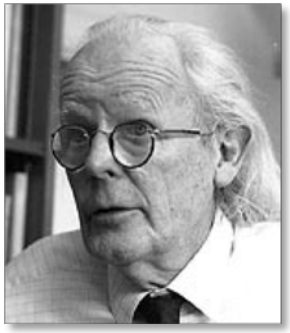
GAME THEORY

Conflict (and cooperation) in multi-agent systems:
Evolutionary games, social welfare, optimization concepts

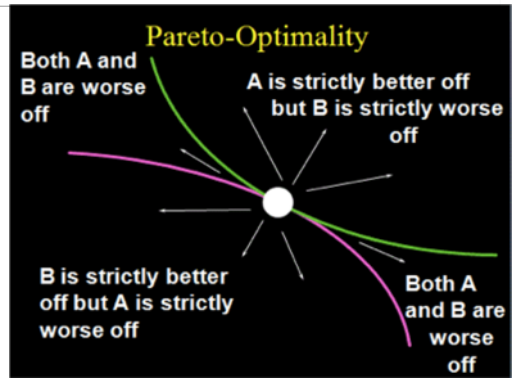


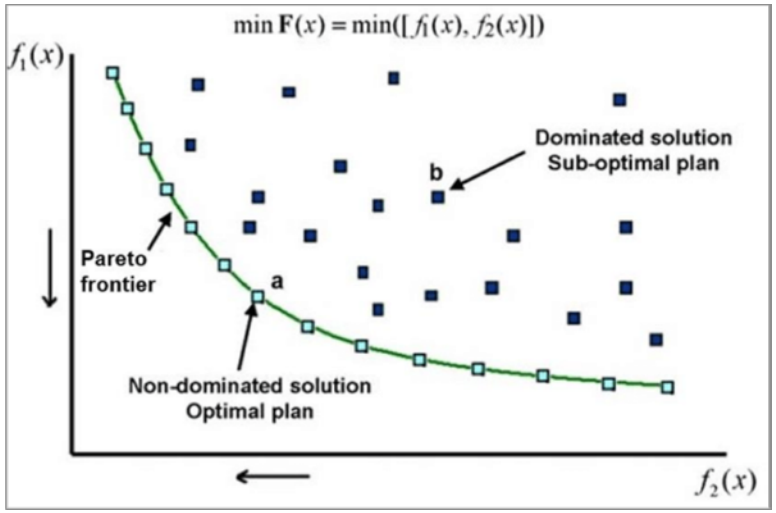
(Gains of player 1, Gains of player 2)	Player 2 acts like... 	
Player 1 acts like... 	(-2,-2)	(2,0)
	(0,2)	(1,1)

J. Maynard Smith



V. Pareto

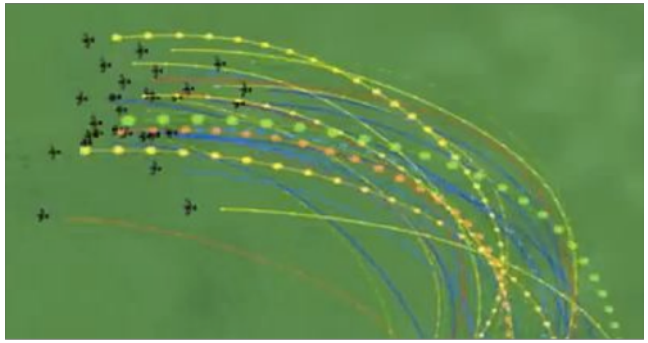




Pattern Formation

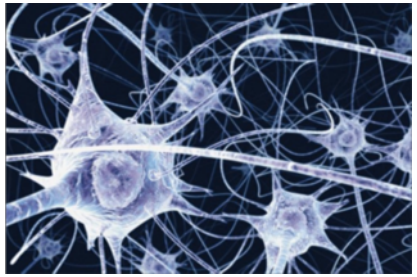
Mobile agents in crowded and/or highly constrained scenarios

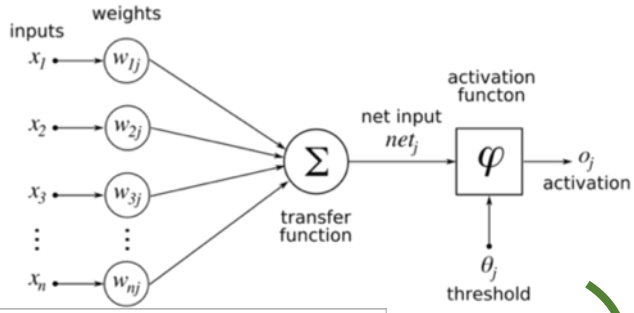
Emergence of order: flows and structures

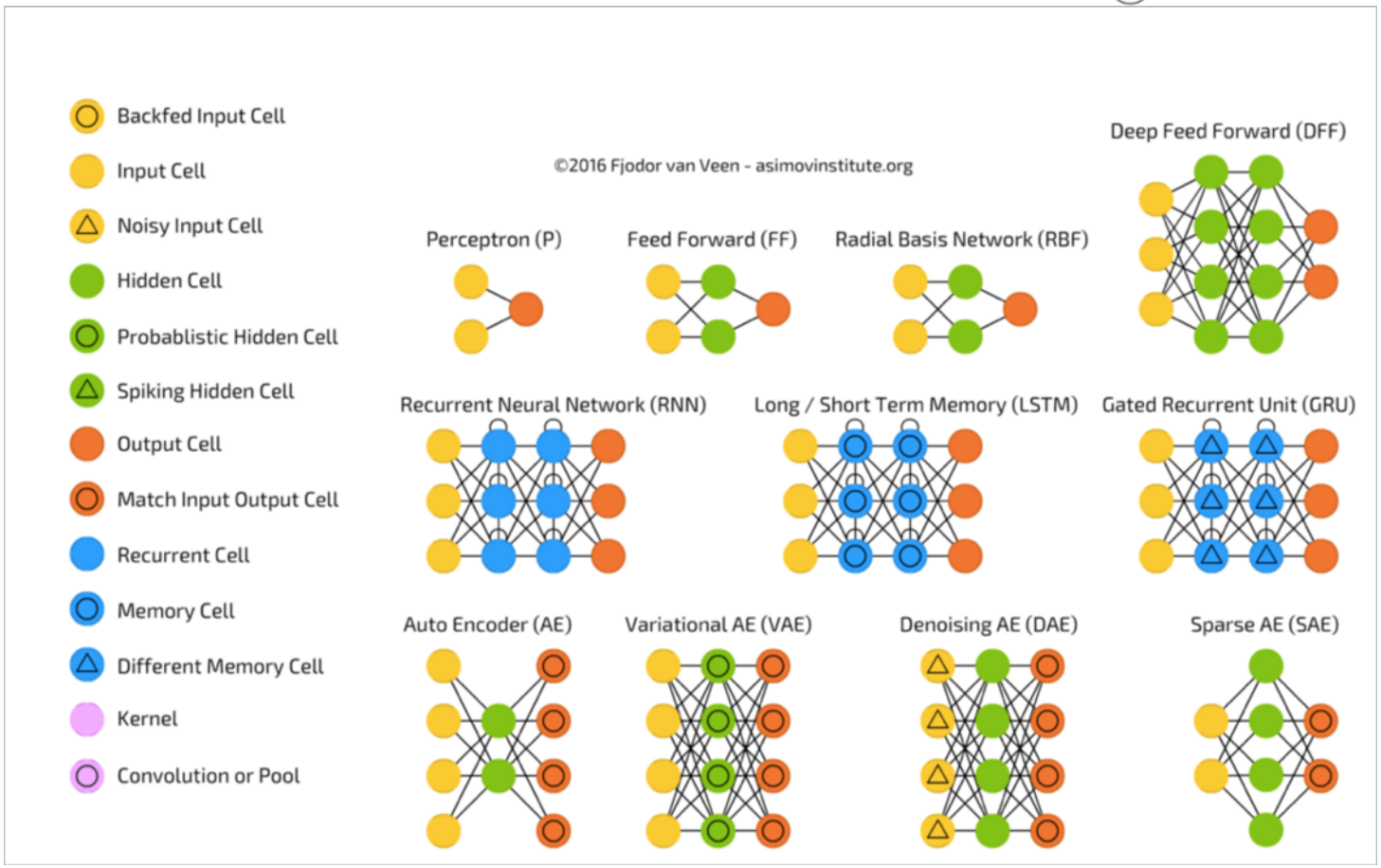


Flocking
Topology / Formation maintenance

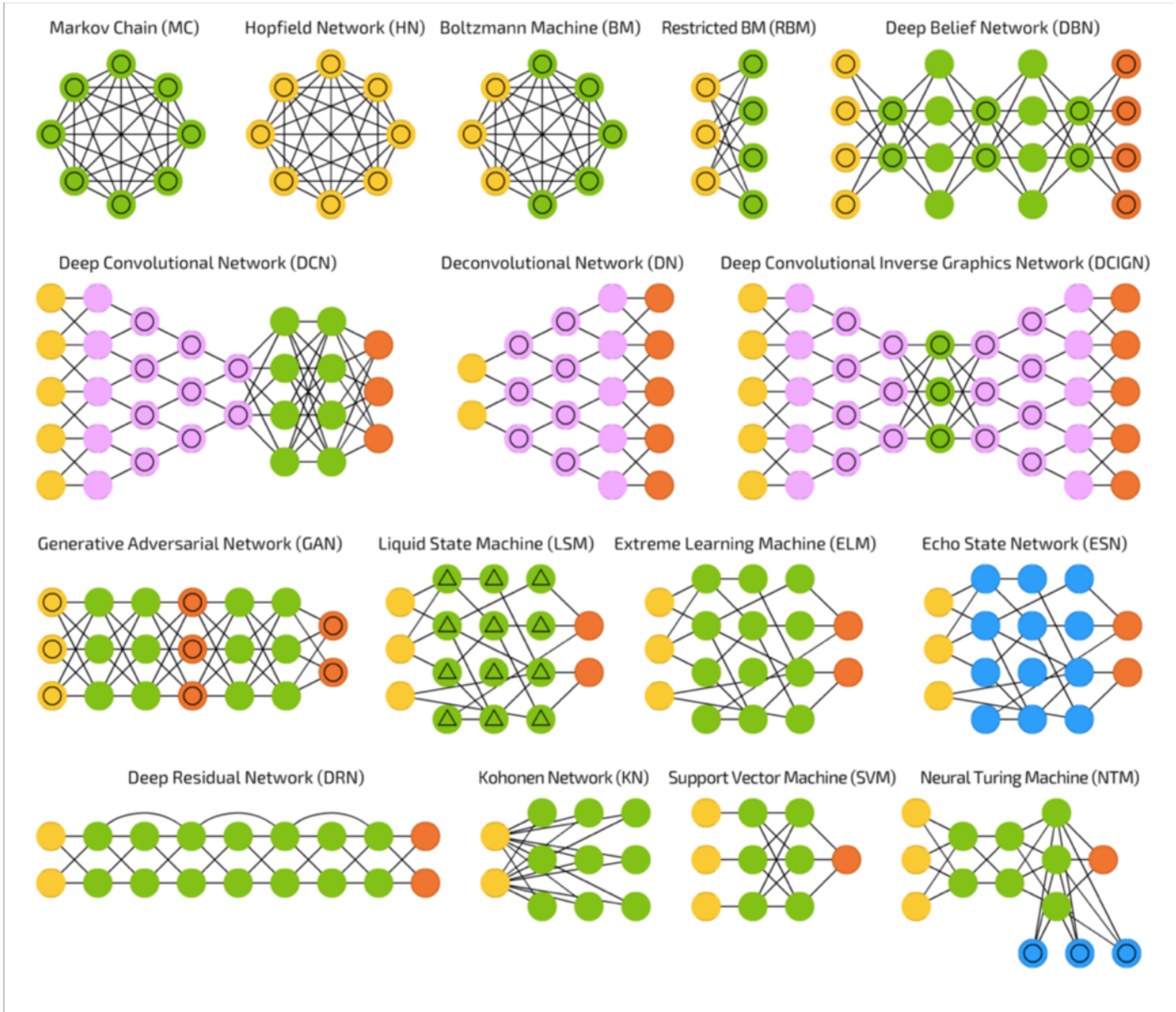
Neural Models







Neural Models



Course Organization

- Two main lectures + one recitation lecture that will be used in various ways (to complete main lectures, to make exercises, to read papers together…)
- Weekly (almost) homework: reading and reviewing papers, answering teoretical and conceptual questions, implement models and algorithms and play (i.e., explore) with them
- 55% of grading will be based on homework
- 15% of grading will result from a midterm exam (theory, concepts, math)
- 30% of grading will come from a final project that will focus on at least two of the topics discussed during the course: select and read a few related papers, work out and implement the main ideas but in a (slightly) different scenario, add something new either in the 'mechanisms' and/or in the analysis
- The final project can be done in group(s) and will be evaluated based on individual reports and oral presentation
- During lectures, be *alive*! Interact with the instructor, ask questions ...

Course Objectives

- Students who successfully complete the course will have acquired **general knowledge of multi-component, complex adaptive systems** in terms of:
  - Challenges related to *modeling, prediction*, and *control* aspects,
  - *Practice* of effectively implementing such systems.
- By studying (complex) systems related to different domains (engineering, biology, economy, robotics, operations research) the student will be exposed to a truly **interdisciplinary background** of mathematical models and application problems.

- You will acquire foundational knowledge and programming practice in a number of important scientific domains, including:
  - Dynamical systems
  - Complex systems
  - Networks
  - Game theory
  - Social choice
  - Swarm intelligence
  - Distributed decision-making
  - Neural networks