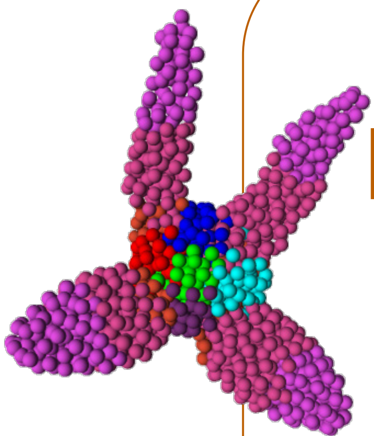
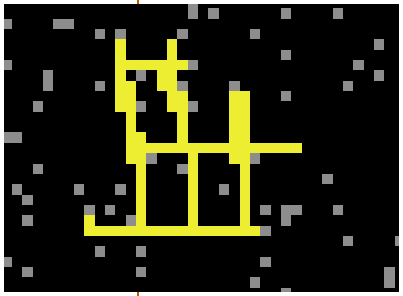
Frontiers of Natural Computing
University of York
York, UK
10th – 12th September 2012



Complex Systems and Morphogenetic Engineering: New Avenues Toward Self-Organized Architecture



~ A Position (60%) and an Illustration (40%) ~

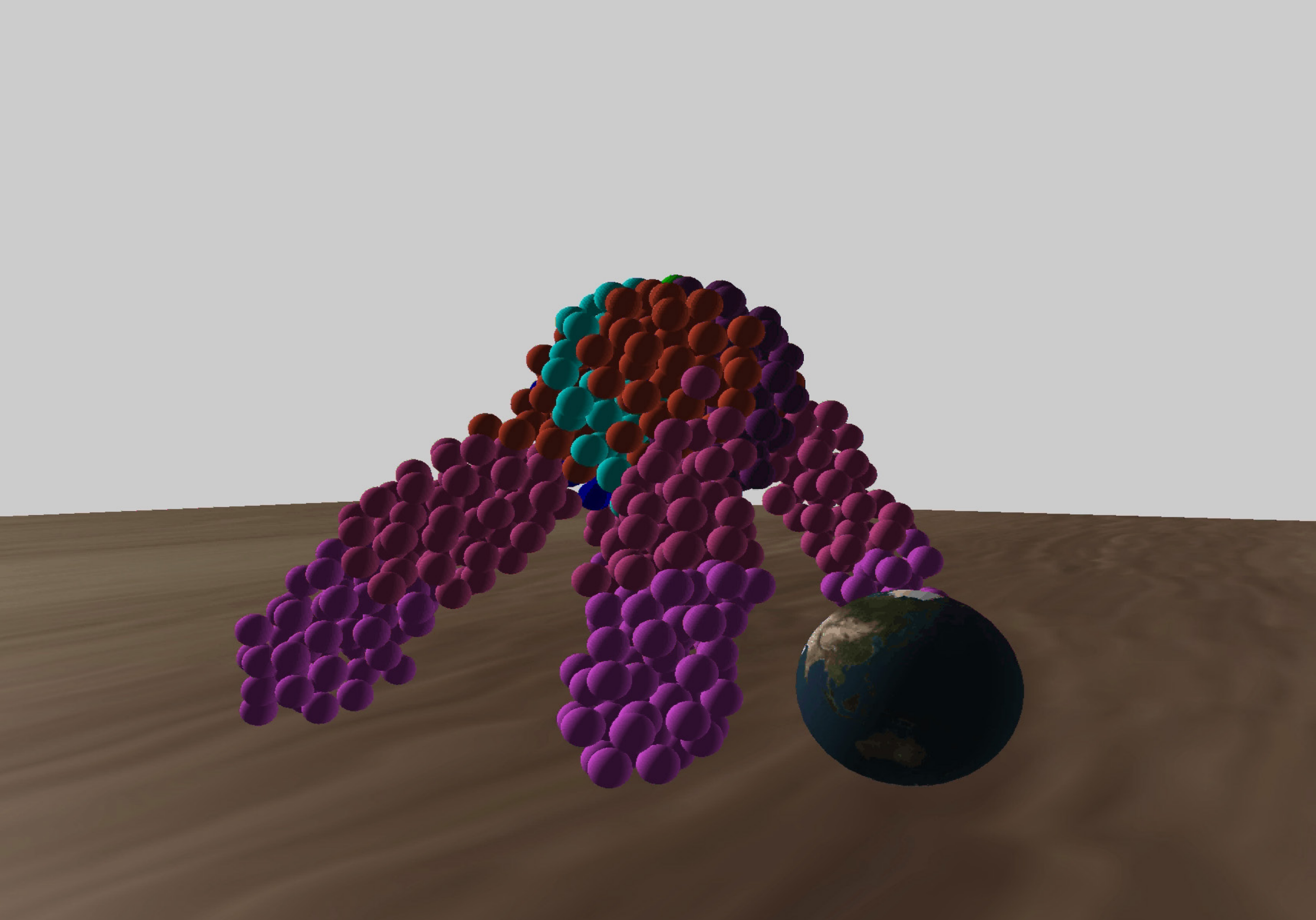
René Doursat

Complex Systems Institute Paris, CNRS / CREA, Ecole Polytechnique
Research Group in Biomimetics, Universidad de Malaga, Spain

INSTITUT DES SYSTEMES COMPLEXES

PARIS ÎLE DE FRANCE

grupo de estudios en biomimética



http://doursat.free.fr/mapdevo.html

Doursat, Sanchez, Fernandez, Kowaliw & Vico (2012)

Susan Stepney, York

+ Stanislaw Ulam [said] that using a term like nonlinear science is like referring to the bulk of **zoology as the study of non-elephant animals**.
+ The elephant in the room here is the classical Turing machine. "**Unconventional**" computation is a similar term: **the study of non-Turing computation**.
+ The **classical Turing machine** was developed as an **abstraction of** how **human "computers"**, **i.e. clerks** following predefined and prescriptive rules, calculated various mathematical tables...
+ [In contrast] unconventional computation can be inspired by the whole of wider nature. We can look to physics (...), to chemistry (reaction-diffusion systems, complex chemical reactions, DNA binding), and to biology (bacteria, flocks, social insects, evolution, growth and self-assembly, immune systems, neural systems), to mention just a few.
+ **PARALLELISM – INTERACTION – NATURE**

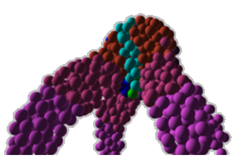
= *COMPLEX SYSTEMS & SELF-ORG.*

The Promise of Unconventional / Natural Computing

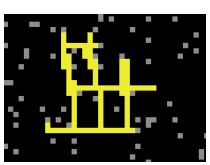
1. Tur(n)ing to Complex Systems (CS)
2. Engineering & Control of Self-Organization (ECSO)
3. Morphogenetic Engineering (ME)
   i.e. "Architectures without Architects"

Position
Illustration

4. Embryomorphic Engineering (EE)
   i.e. "Artificial Life Evo-Devo"



4.a. *MapDevo*
Modular Architecture by Programmable Development



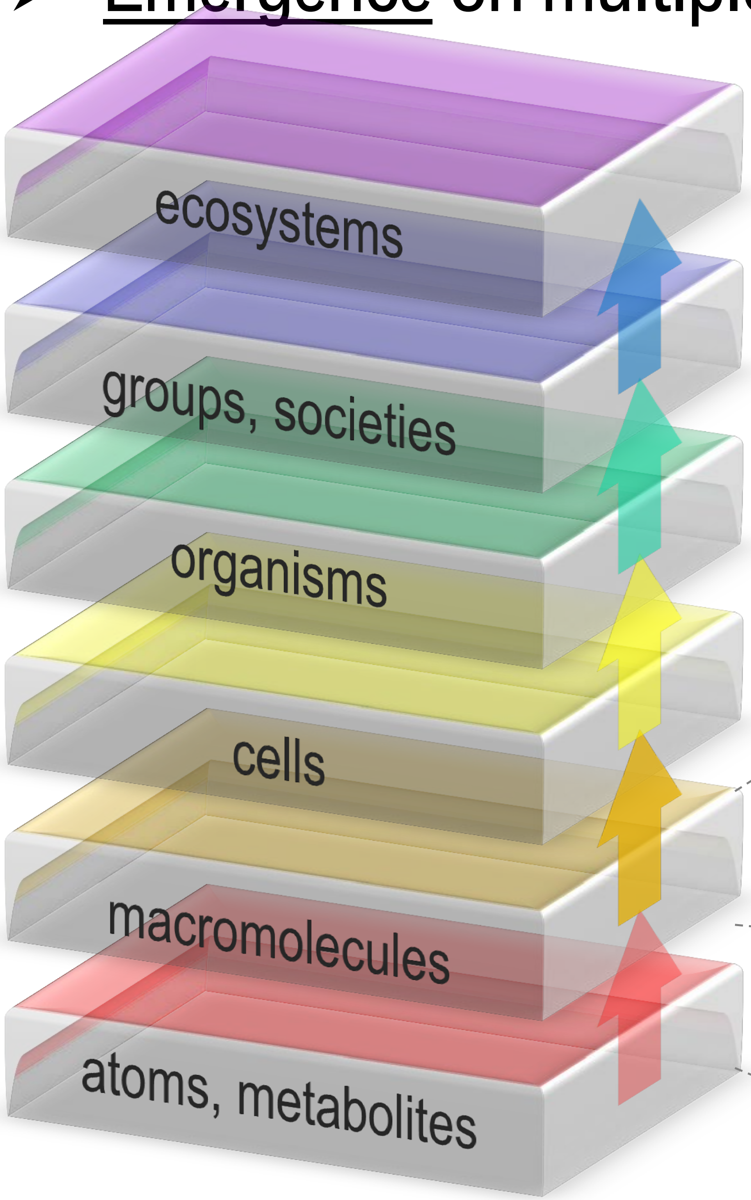
4.b. *ProgLim*
Program-Limited Aggregation

The Promise of Unconventional / Natural Computing

1. Tur(n)ing to Complex Systems (CS)

1. Turning to Complex Systems

- Emergence on multiple levels of self-organization

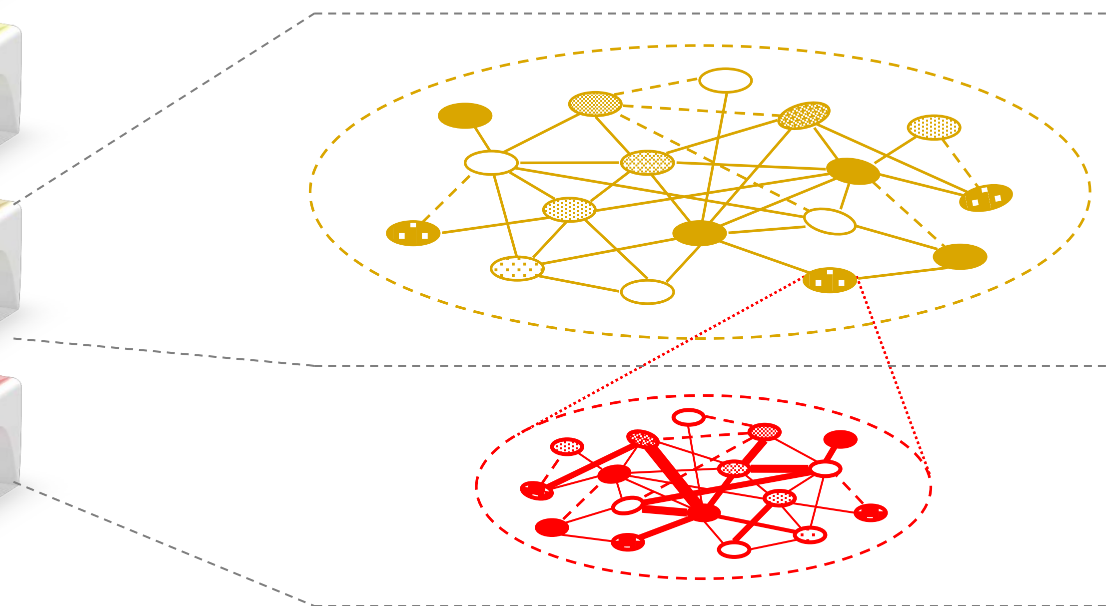


complex systems:

a) a large number of elementary agents interacting locally

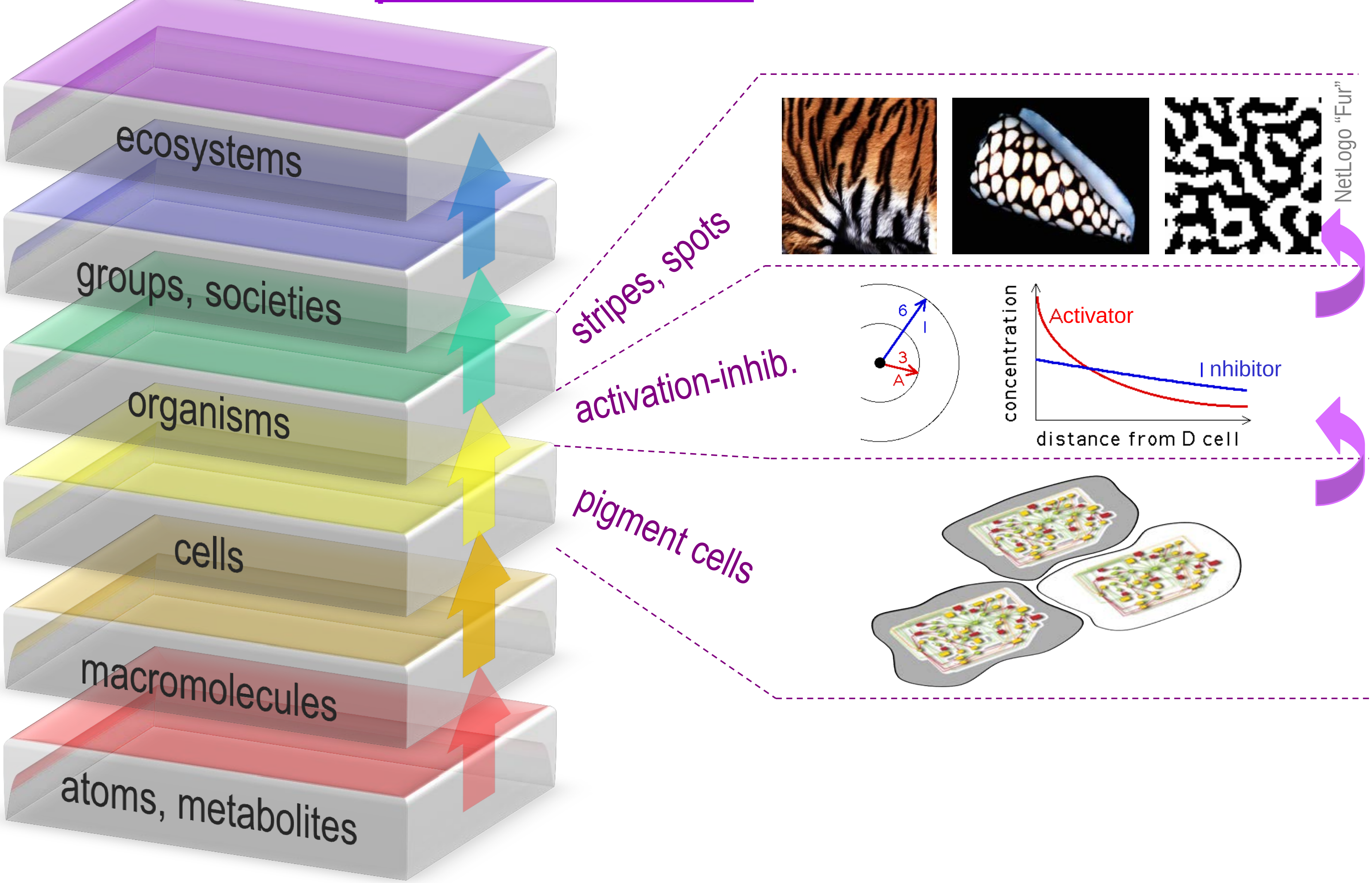
b) simple individual behaviors creating a complex emergent collective behavior

c) decentralized dynamics: no master blueprint or grand architect



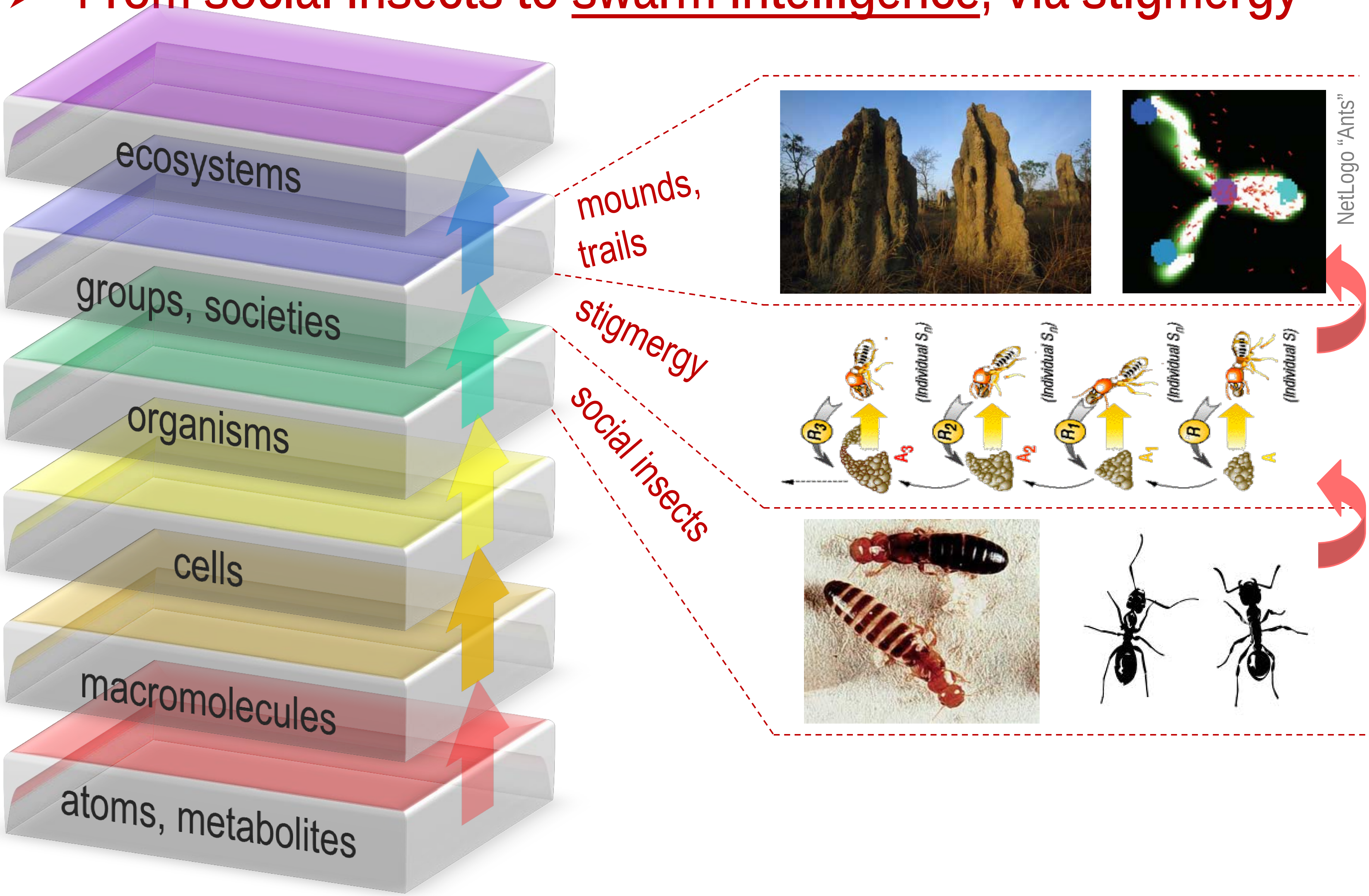
1. Turning to Complex Systems

➢ From cells to <u>pattern formation</u>, via reaction-diffusion



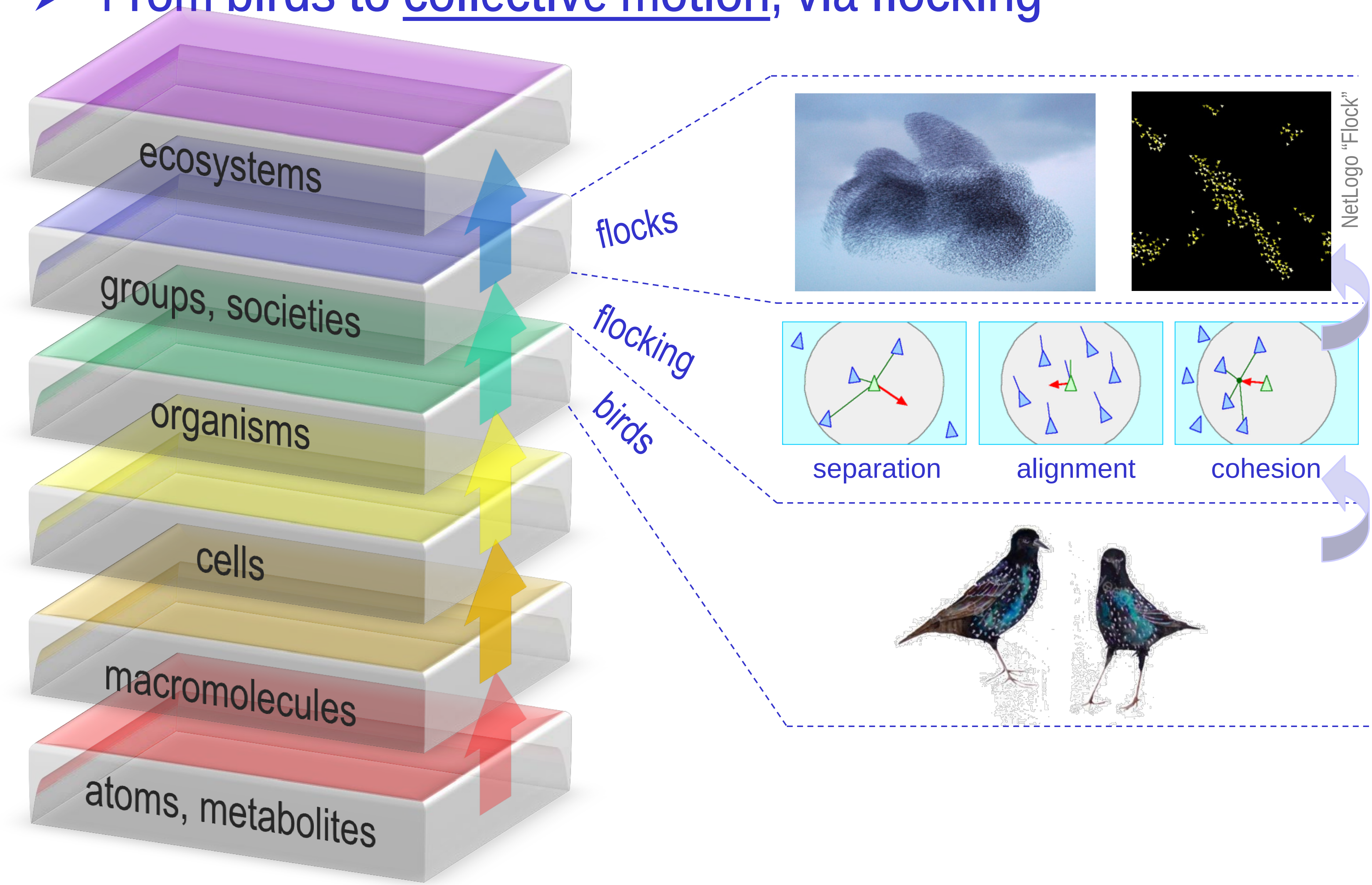
1. Turning to Complex Systems

- From social insects to <u>swarm intelligence</u>, via stigmergy



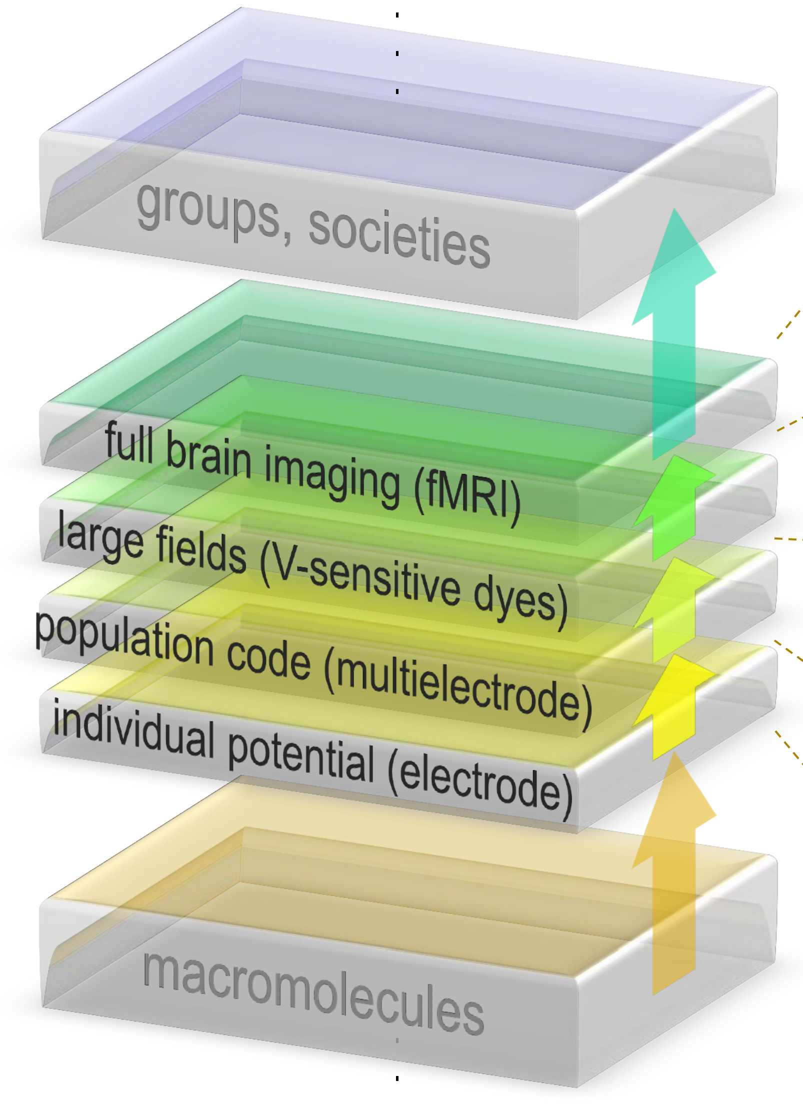
1. Turning to Complex Systems

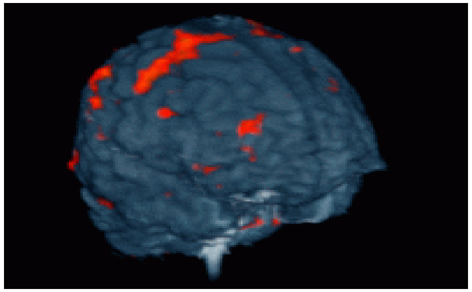
➢ From birds to collective motion, via flocking



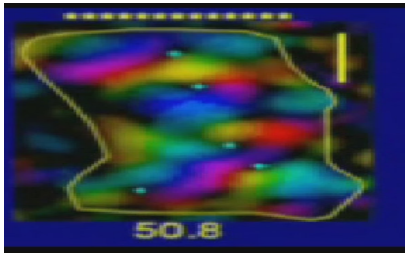
1. Turning to Complex Systems

➢ From neurons to <u>cognition</u>, via synaptic transmission

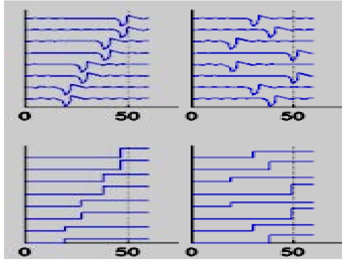




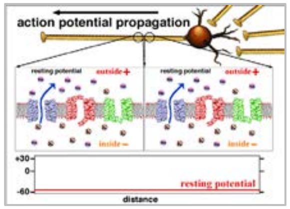
Animation of a functional MRI study (J. Ellermann, J. Strupp, K. Ugurbil, U Minnesota)



Dynamics of orientation tuning: polar movie Sharon and Grinvald, Science 2002

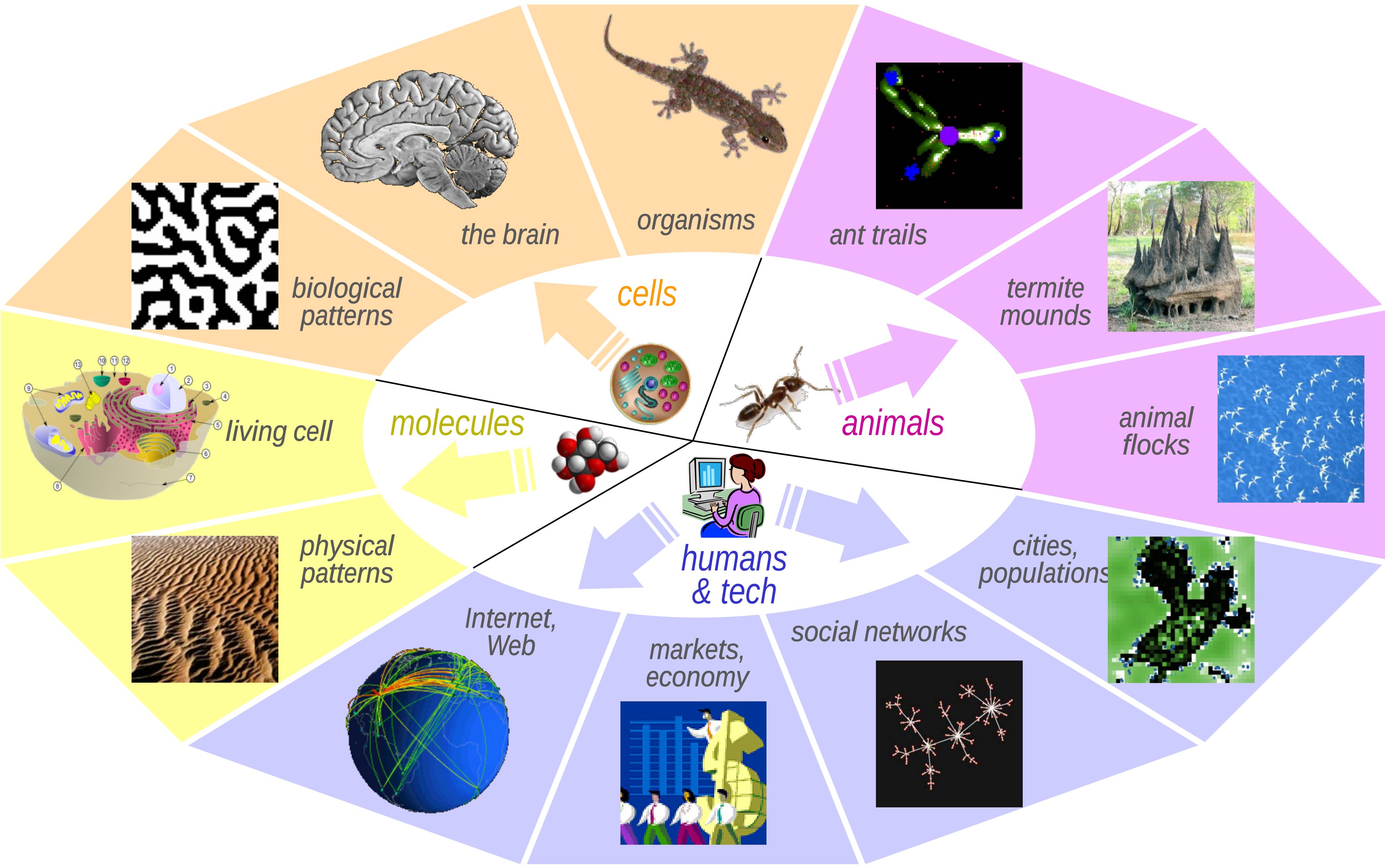


Raster plot of of a simulated synfire braid, Doursat et al. 2011



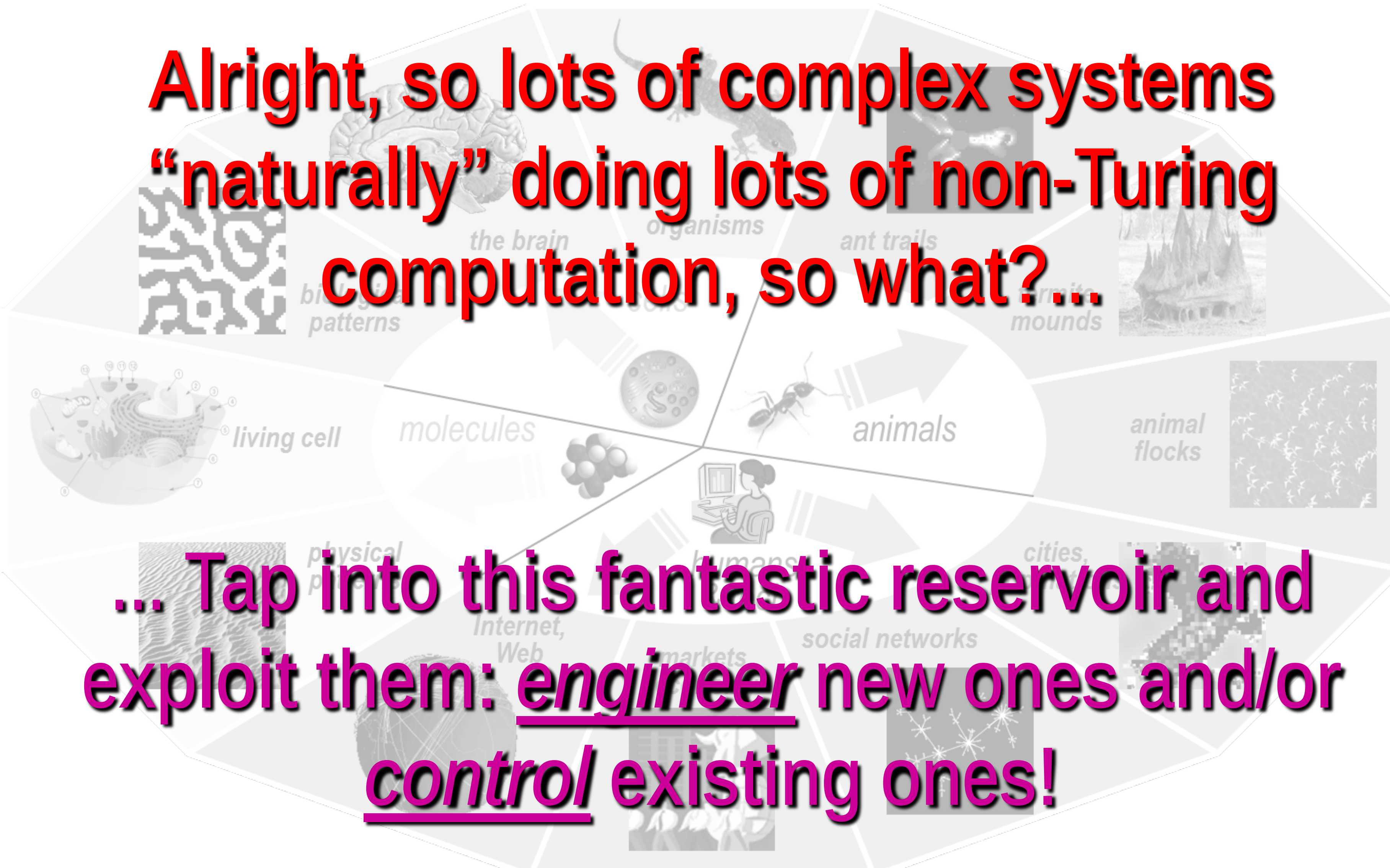
1. Turning to Complex Systems

- All agent types: molecules, cells, animals, humans & tech



1. Turning to Complex Systems

- All agent types: molecules, cells, animals, humans & tech



Alright, so lots of complex systems "naturally" doing lots of non-Turing computation, so what?...

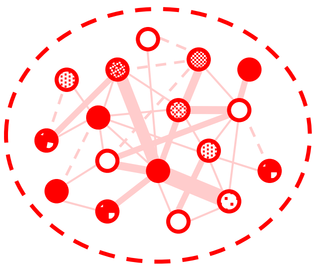
... Tap into this fantastic reservoir and exploit them: *engineer* new ones and/or *control* existing ones!

The Promise of Unconventional / Natural Computing

1. Tur(n)ing to Complex Systems (CS)
2. Engineering & Control of Self-Organization (ECSO)

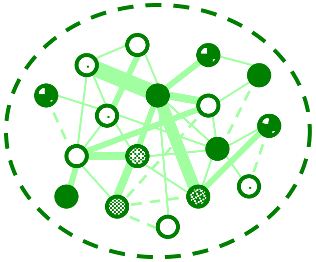
2. Engineering & Control of Self-Organization

➢ Between natural and engineered emergence



CS Science: observing and understanding "natural", spontaneous emergence (including human-caused)
→ *Agent-Based Modeling (ABM)*

Unconventional / Natural CS Computation:
fostering <u>and</u> guiding complex systems at the level of their elements



CS (ICT) Engineering: creating and programming a new, artificial self-organization / emergence
→ *(Complex) Multi-Agent Systems (MAS)*

2. Engineering & Control of Self-Organization

- Exporting models of natural CS to ICT: "(bio-)inspiration"
  - already a tradition...

ex: neurons & brain	ex: ant colonies / bird flocks	ex: genes & evolution
biological neural models	trails, swarms / collective motion	laws of genetics
binary neuron, linear synapse	move, deposit, follow "pheromone" / separation, alignment, cohesion ("boids")	genetic program, binary code, mutation
artificial neural networks (ANNs) applied to machine learning & classification	ant colony optimization (ACO) graph theoretic & networking problems / particle swarm optimization (PSO) "flying over" solutions in high-D spaces	genetic algorithms (GAs), & evolutionary computation for search & optimization

TODAY: simulated in a Turing machine / von Neumann architecture

2. Engineering & Control of Self-Organization

- Exporting models of natural CS to ICT: "(bio-)inspiration"
  - ✓ ... and looping back onto an unconventional physical implementation to fully exploit the "in materio" computational efficiency

DNA computing

chemical, wave-based computing

swarm robotics

synthetic biology

artificial neural networks (ANNs) applied to machine learning & classification

ant colony optimization (ACO) graph theoretic & networking problems / particle swarm optimization (PSO) "flying over" solutions in high-D spaces

genetic algorithms (GAs), & evolutionary computation for search & optimization

TOMORROW: running in truly parallel roboware, bioware, nanoware, etc.

2. Engineering & Control of Self-Organization

- Exporting models of natural CS to ICT: "(bio-)inspiration"
  - ✓ common shortcoming: the classical "(over-)engineering" attitude
    - reintroducing too much **exogenous**, top-down design/control
      - → *exploit and only "influence" the natural endogenous properties!*
    - **breaking** things apart too much
      - → *keep the system internally complex and self-organized!*
    - the **stepwise** fallacy: start with simple tasks (ex: XOR) and build up from there
      - → *go for the collective attractors right away! (Kauffman's "order for free")*
    - keeping self-organization, but **forcing** it into rigid design (logic gate <u>flow</u> ≠ RBN)
    - clinging to computing **universality**
      - → *specialization is more promising!*

specific natural or societal <u>complex system</u>

↓

model simulating this system

↓

generic principles and mechanisms (schematization, caricature)

↓

<u>new computational discipline</u> exploiting these principles to solve ICT problems

2. Engineering & Control of Self-Organization

➢ Ex. of "over-engineered" bio-inspired domain: evo computation

John Holland, founder of GAs <u>and</u> major promoter of Complex (Adaptive) Systems: his titles → ~100% CAS

- *Hidden order: How adaptation builds <u>complexity</u>*
- *<u>Emergence</u>: From chaos to order*
- *Artificial adaptive <u>agents</u> in economic theory*
- *Outline for a logical theory of adaptive <u>systems</u>*
- *Studying <u>complex adaptive systems</u>*
- *Exploring the evolution of complexity in <u>signaling networks</u>*
- *Can there be a unified theory of <u>CAS</u>?*
- *etc.*

GECCO, among the biggest (and most selective) EC conferences: its tracks in 2009 → ~15% CAS?

(... individuals internally complex?)

- *<u>ACO and Swarm Intelligence</u>*
- *<u>Artificial Life, Evolutionary Robotics</u>, Adaptive Behavior, Evolvable Hardware*
- *Bioinformatics and Computational Biology*
- *Combinatorial Optimization and Metaheuristics*
- *Estimation of Distribution Algorithms*
- *Evolution Strategies and Evolutionary Programming*
- *Evolutionary Multiobjective Optimization*
- *<u>Generative and Developmental Systems</u>*
- *Genetic Algorithms*
- *Genetic Programming*
- *Genetics-Based Machine Learning*
- *Parallel Evolutionary Systems*
- *Real World Application*
- *Search Based Software Engineering*

2. Engineering & Control of Self-Organization

- Other examples of “over-engineered” bio-inspired domains:
  - artificial neural networks (IJCNN, etc.)
  - multi-agent systems (AAMAS, etc.)
  - ... even artificial life! (Alife, ECAL) is playing down the self-organization and complex systems properties of living matter
- Conversely: “natural” complex systems conferences don’t show much concern for design, engineering or control issues
  - overcrowded with (statistical) physicists (ICCS, ECCS, etc.)
  - or overcrowded with biologists

... and that's why we need

ECSO 2014

1st International Conference on the
Engineering & Control of Self-Organization

Home | Call for Papers | Submission | Program | Organization | Registration | Venue

(dates and venue to be announced)

Overview

This conference is the new center for all the emerging cross-disciplinary topics that attempt to solve, at their core, the paradox of *engineering and controlling self-organization* (ECSO) in various forms, and for this reason cannot find an appropriate home in traditional venues. ECSO researchers are torn between scientific domains focused on the pure observation and modeling of natural complex systems (in which they appear too "artificial") and engineering domains more interested in top-down design and optimization than complexity (in which they appear too "soft" or "bio-inspired").

- Amorphous Computing, Spatial Computing
- Morphogenetic Engineering, Embryomorphic Engineering
- Morphogenetic Robotics, Evo-Devo Robotics, Generative Robotics
- Swarm Robotics, Collective Robotics, Swarm Intelligence, Swarm Chemistry
- Pervasive Computing, Ubiquitous Computing, Ambient Intelligence
- Pervasive Adaptation, Autonomic Computing, Autonomic Networking, Emergent Engineering
- Organic Computing
- Artificial Development, Artificial Embryogeny, Artificial Ontogeny, Neuroevolution
- Generative Systems, Rewriting Systems
- Unconventional Computing, Natural Computing
- Complex Systems Engineering
- Synthetic Biology, Biological and Chemical IT, Artificial Chemistry
- Multi-Agent Systems, Agent-Based Modeling

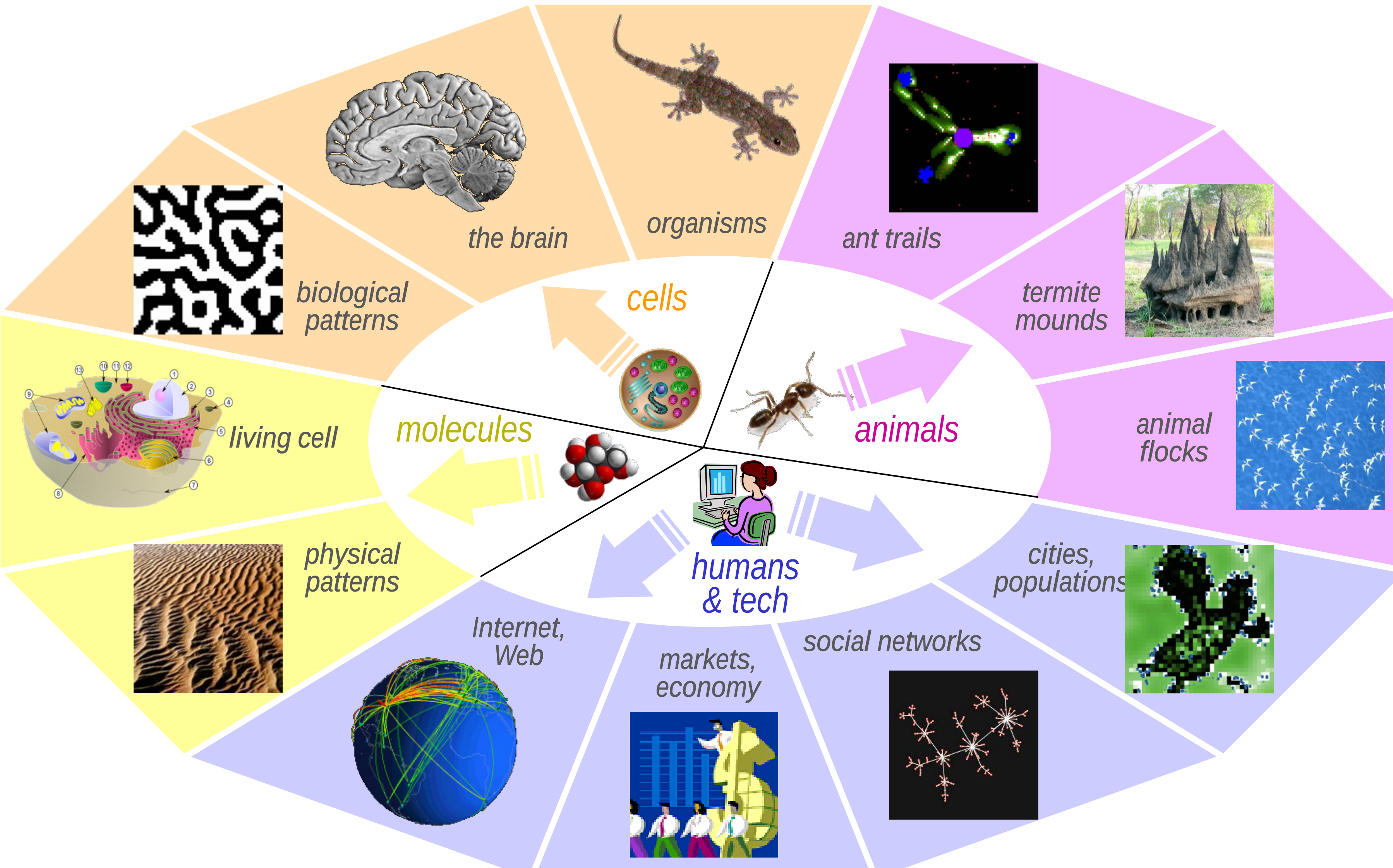
http://iscpif.fr/ecso2014

The Promise of Unconventional / Natural Computing

1. Tur(n)ing to Complex Systems (CS)
2. Engineering & Control of Self-Organization (ECSO)
3. Morphogenetic Engineering (ME)
   i.e. "Architectures without Architects"

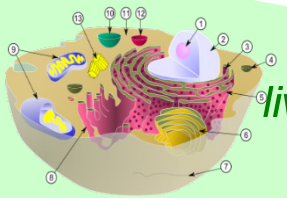
3. Architectures Without Architects

➢ All agent types: molecules, cells, animals, humans & tech



3. Architectures Without Architects

- "Simple"/random vs. architectured complex systems



- biology strikingly demonstrates the possibility of **combining** pure self-organization and elaborate architecture, i.e.:
  - ✓ a non-trivial, sophisticated morphology
    - *hierarchical* (multi-scale): regions, parts, details
    - *modular*: reuse of parts, quasi-repetition
    - *heterogeneous*: differentiation, division of labor
  - ✓ *random* at agent level, *reproducible* at system level

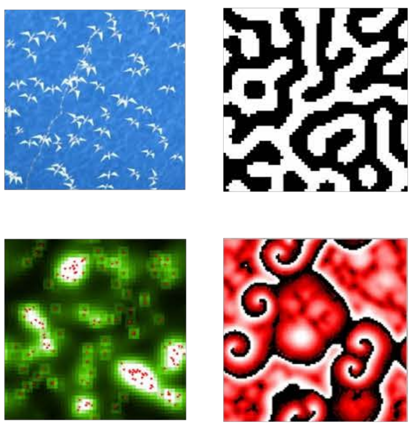
3. Morphogenetic Engineering

Artificial Systems

Natural Systems

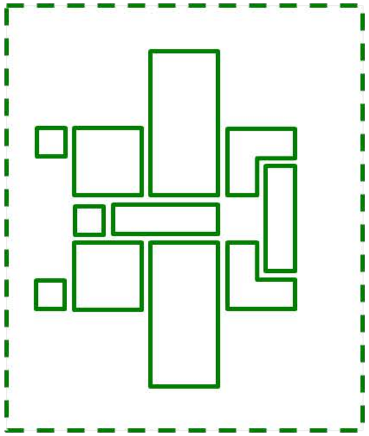
(d)

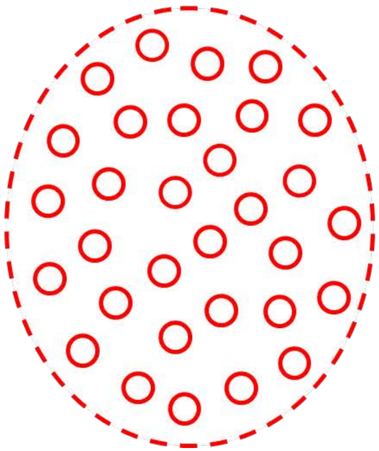
(a)



Architectured Systems Showing *no* Self-Organization

Self-Organized Systems Showing *no* Architecture





http://doursat.free.fr/morpheng.html

Doursat, Sayama & Michel (2012)

3. Morphogenetic Engineering

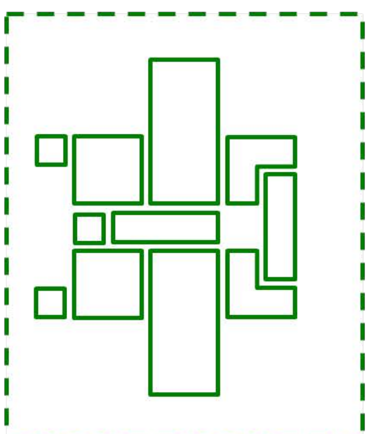
Artificial Systems

more architecture,
less randomness

Natural Systems

(d)

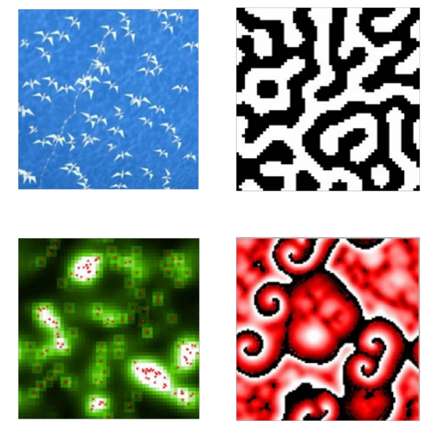
Architectured Systems
Showing *no* Self-Organization



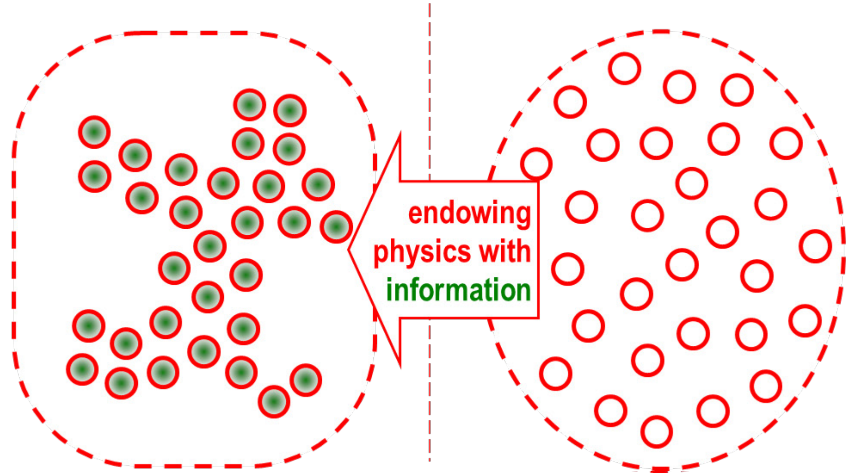
(b)

Self-Organized Systems
Showing an Architecture

(a)



Self-Organized Systems
Showing *no* Architecture



Doursat, Sayama & Michel (2012)

3. Morphogenetic Engineering

Artificial Systems — more self-organization, less design →

← Natural Systems

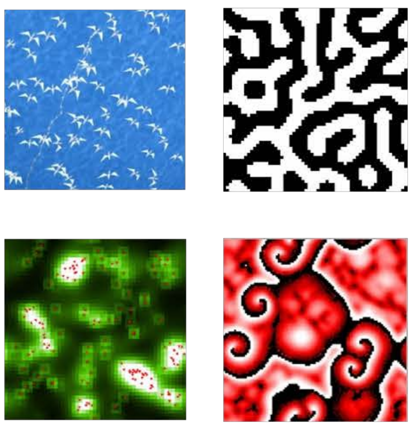
(d)

Architectured Systems Showing *no* Self-Organization

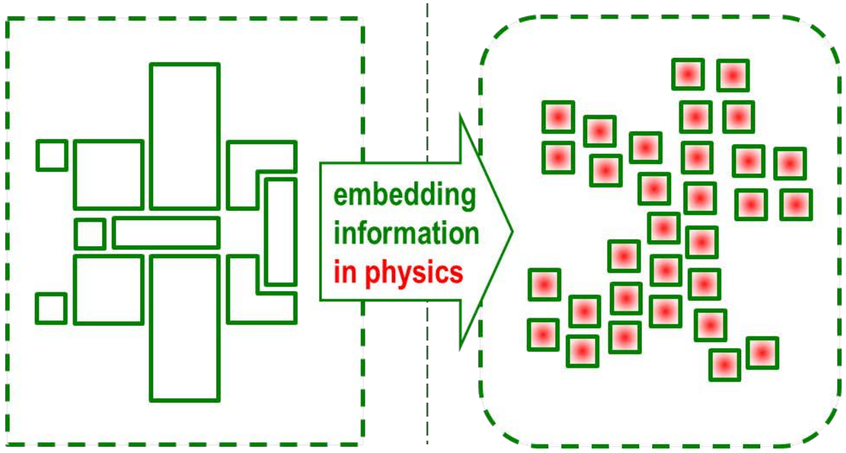
(c)

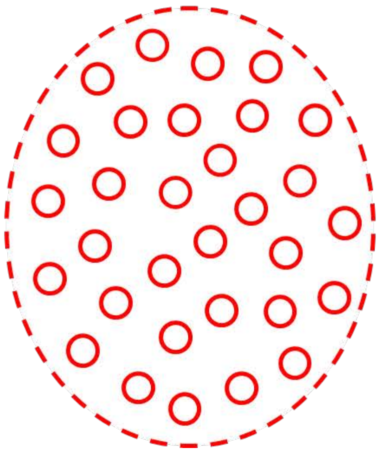
Architectured Systems Showing Self-Organization

(a)



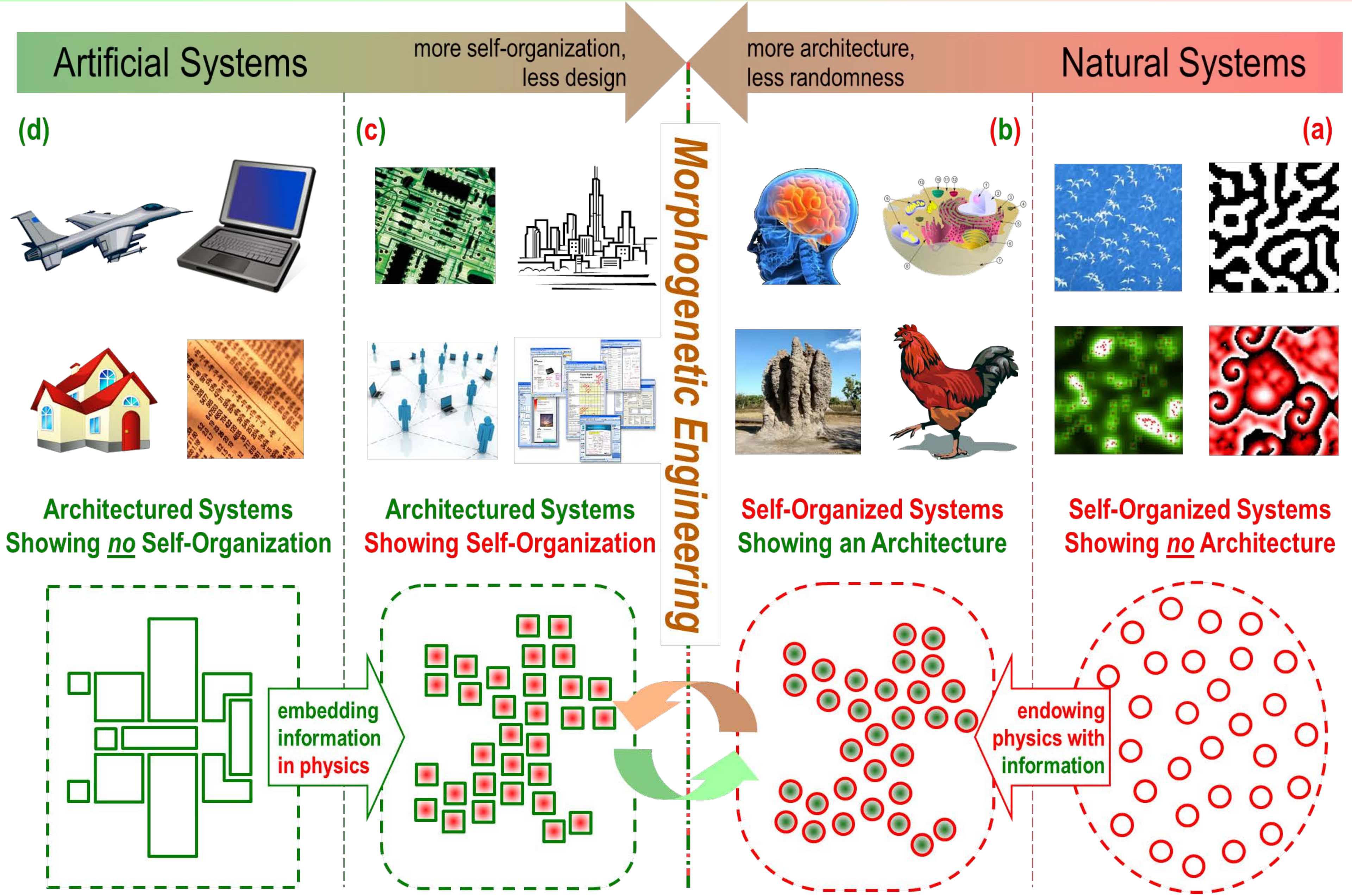
Self-Organized Systems Showing *no* Architecture





Doursat, Sayama & Michel (2012)

3. Morphogenetic Engineering



Doursat, Sayama & Michel (2012)

3. Morphogenetic Engineering

Morphogenetic Engineering (ME) is about designing the agents of self-organized architectures*... not the architectures directly*

- ME brings a new focus inside the ECSO family
  - ✓ exploring the artificial design and implementation of **decentralized systems capable of developing elaborate, heterogeneous morphologies** without central planning or external lead

- Related emerging ICT disciplines and application domains
  - ✓ *amorphous/spatial computing* (MIT, Fr.)
  - ✓ *organic computing* (DFG, Germany)
  - ✓ *pervasive adaptation* (FET, EU)
  - ✓ *ubiquitous computing* (PARC)
  - ✓ *programmable matter* (CMU)
  - ✓ swarm robotics, modular/reconfigurable robotics
  - ✓ mobile ad hoc networks, sensor-actuator networks
  - ✓ synthetic biology, etc.

3. Morphogenetic Engineering

1st Morphogenetic Engineering Workshop, ISC, Paris 2009
http://iscpif.fr/MEW2009

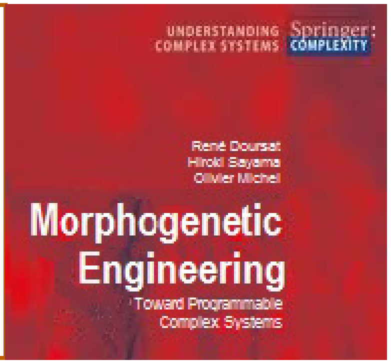
2nd Morphogenetic Engineering Session, ANTS 2010, Brussels
http://iridia.ulb.ac.be/ants2010

3rd Morphogenetic Engineering Workshop, ECAL 2011, Paris
http://ecal11.org/workshops#mew

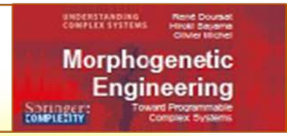
Morphogenetic Engineering:

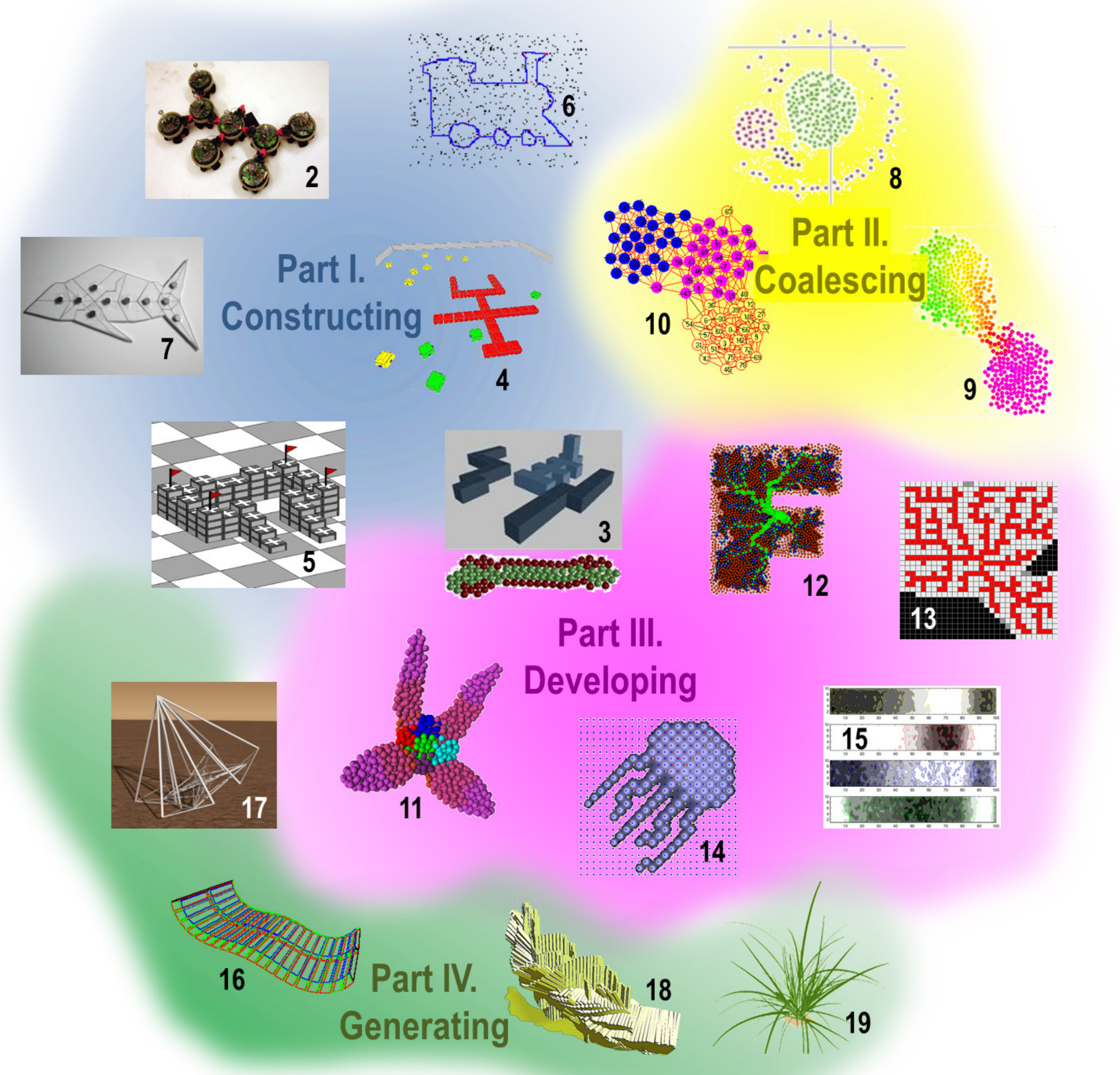
Toward Programmable Complex Systems

R. Doursat, H. Sayama & O. Michel, eds. **Fall 2012, Springer**



Morphogenetic Engineering:
Toward Programmable Complex Systems





Doursat, Sayama & Michel, eds. (2012)

2) O'Grady, Christensen & Dorigo

3) Jin & Meng

4) Liu & Winfield

5) Werfel

6) Arbuckle & Requicha

7) Bhalla & Bentley

8) Sayama

9) Bai & Breen

10) Nembrini & Winfield

11) Doursat, Sanchez, Dordea, Fourquet & Kowaliw

12) Beal

13) Kowaliw & Banzhaf

14) Cussat-Blanc, Pascalie, Mazac, Luga & Duthen

15) Montagna & Viroli

16) Michel, Spicher & Giavitto

17) Lobo, Fernandez & Vico

18) von Mammen, Phillips, Davison, Jamniczky, Hallgrimsson & Jacob

19) Verdenal, Combes & Escobar-Gutierrez

The Promise of Unconventional / Natural Computing

1. Tur(n)ing to Complex Systems (CS)
2. Engineering & Control of Self-Organization (ECSO)
3. Morphogenetic Engineering (ME)
   i.e. "Architectures without Architects"
4. Embryomorphic Engineering (EE)
   i.e. "An Artificial Life Evo-Devo"

4. Embryomorphic Engineering: Alife Evo-Devo

- Exporting models of natural CS to ICT: "(bio-)inspiration"
  - already a tradition (although too re-engineered and "decomplexified")

ex: neurons & brain

biological neural models

binary neuron, linear synapse

artificial neural networks (ANNs) applied to machine learning & classification

ex: ant colonies / bird flocks

trails, swarms / collective motion

move, deposit, follow "pheromone" / separation, alignment, cohesion ("boids")

ant colony optimization (ACO) graph theoretic & networking problems / particle swarm optimization (PSO) "flying over" solutions in high-D spaces

ex: genes & evolution

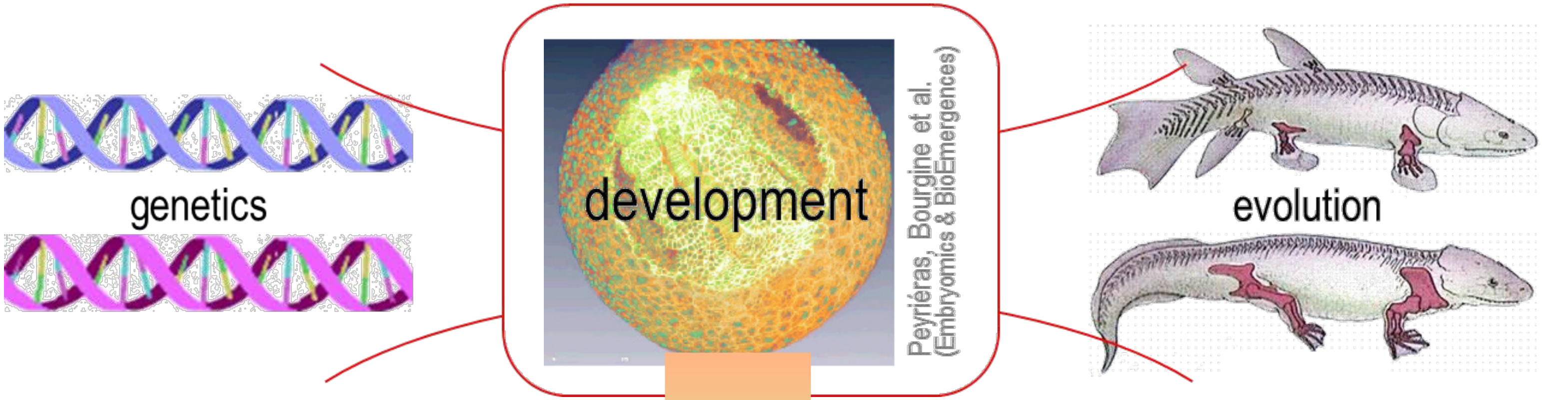
laws of genetics

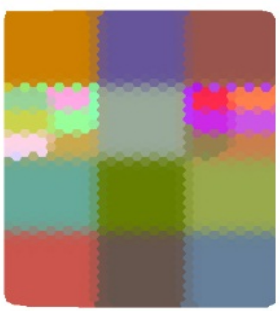
genetic program, binary code, mutation

genetic algorithms (GAs), & evolutionary computation for search & optimization

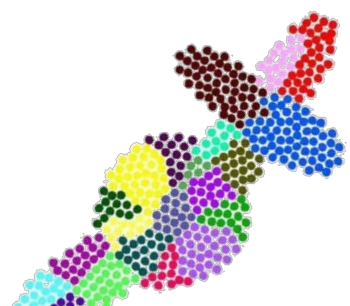
whether simulated in a Turing machine... ... or embedded in bioware, nanoware...

- A new line of bio-inspiration: ***biological morphogenesis***
  - ✓ designing multi-agent models for decentralized systems engineering

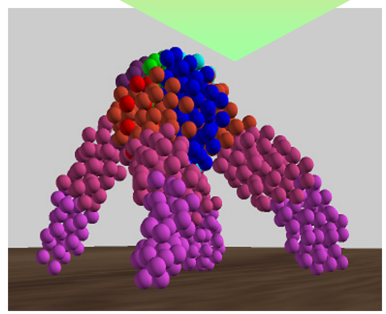




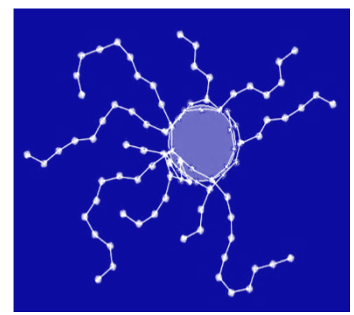
Doursat (2006)



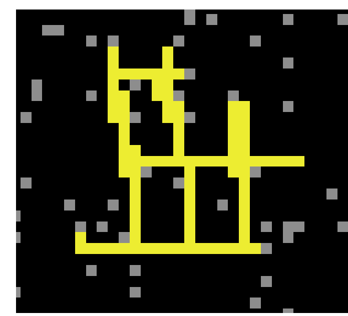
Doursat (2008, 2009)



Doursat, Sanchez, Fernandez Kowaliw & Vico (2012)



Doursat & Ulieru (2009)

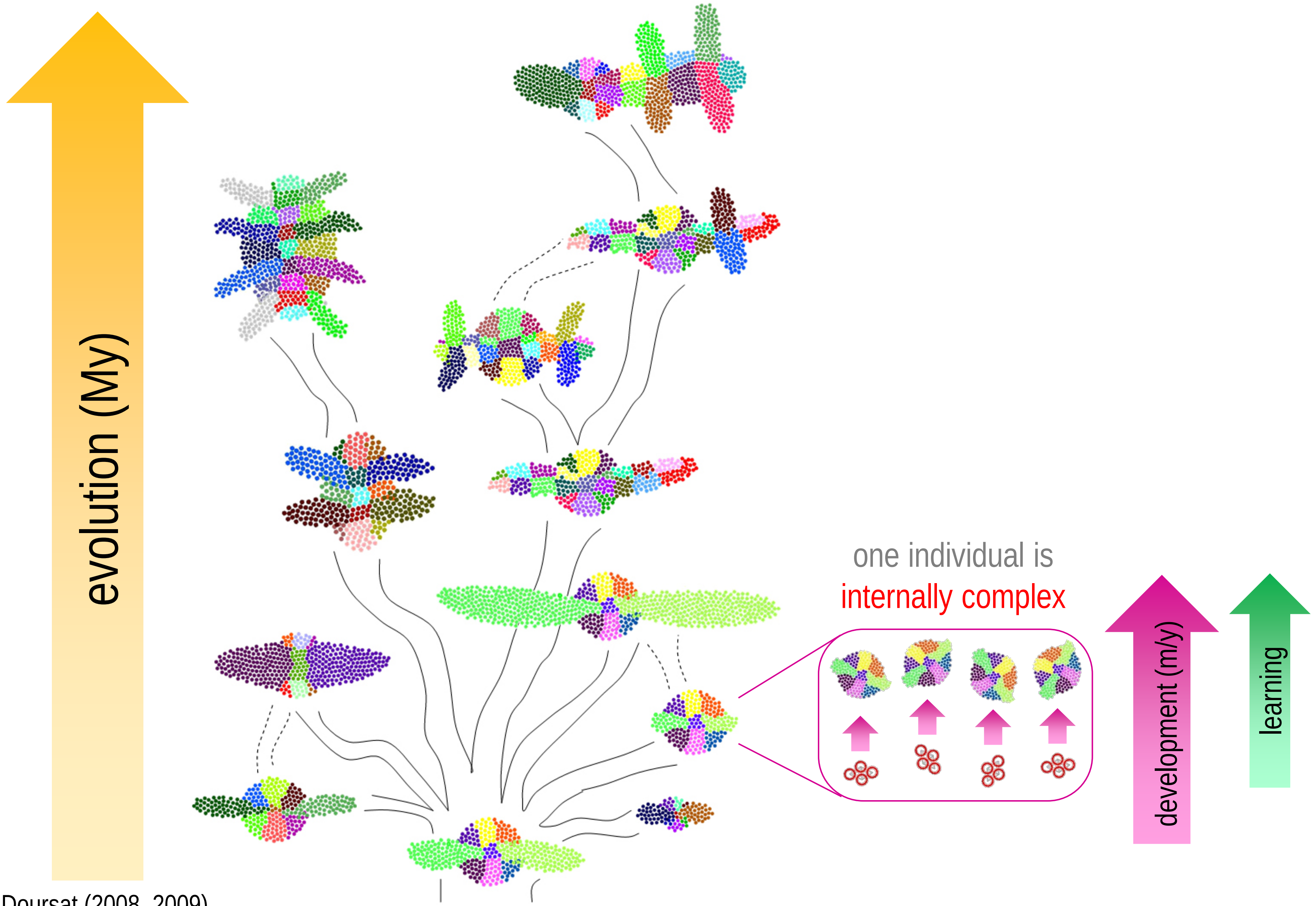


Doursat, Fourquet, Dordea & Kowaliw (2012)

Embryomorphic Engineering

whether simulated in a Turing machine...	... or embedded in bioware, nanoware...

4. Embryomorphic Engineering: Alife Evo-Devo



Doursat (2008, 2009)

4. Embryomorphic Engineering: Alife Evo-Devo

Nothing in biology makes sense except in the light of evolution

—Dobzhansky, 1973

Evo-Devo

Not much in evolution makes sense except in the light of multicellular development

(or molecular self-assembly for unicellular organisms)

Nothing in development makes sense except in the light of complex systems

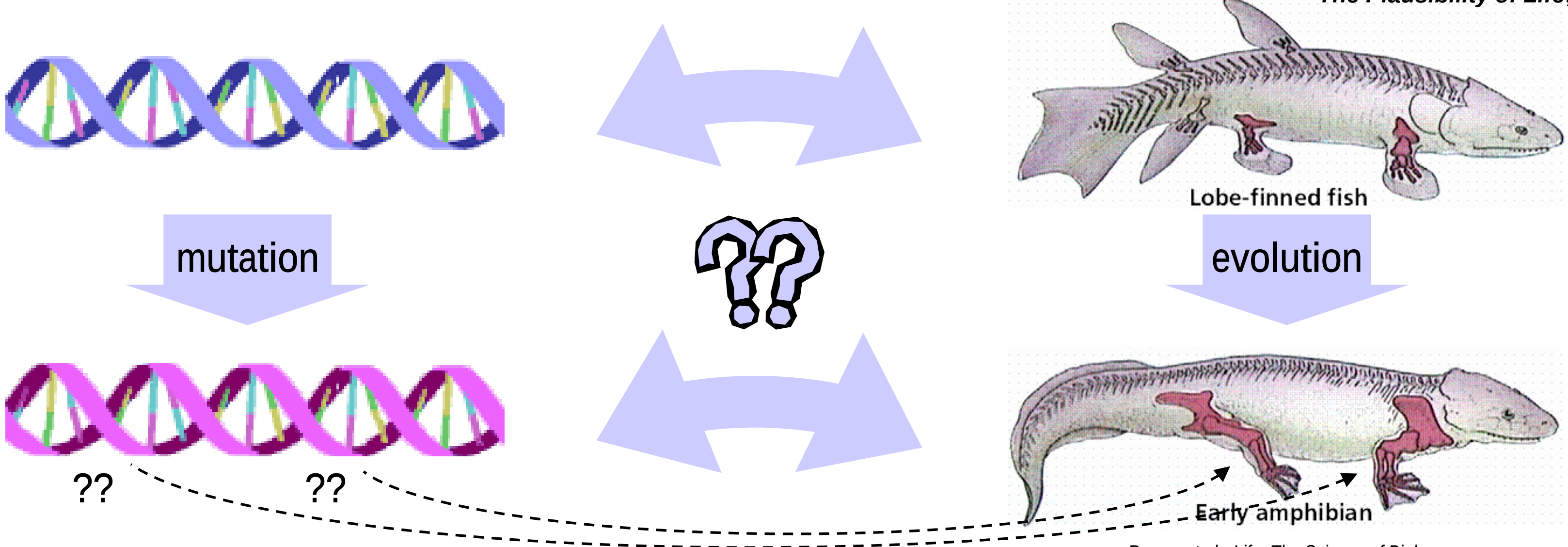
4. Embryomorphic Engineering: Alife Evo-Devo

- Development: the missing link of the Modern Synthesis...

"When Charles Darwin proposed his theory of evolution by variation and selection, explaining selection was his great achievement. He could not explain variation. That was Darwin's dilemma."

"To understand novelty in evolution, we need to understand organisms down to their individual building blocks, down to their deepest components, for these are what undergo change."

—Marc W. Kirschner and John C. Gerhart (2005)
***The Plausibility of Life*, p. ix**



Purves et al., *Life: The Science of Biology*

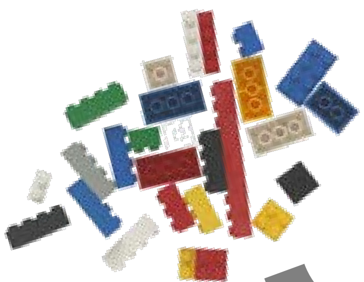
4. Embryomorphic Engineering: Alife Evo-Devo

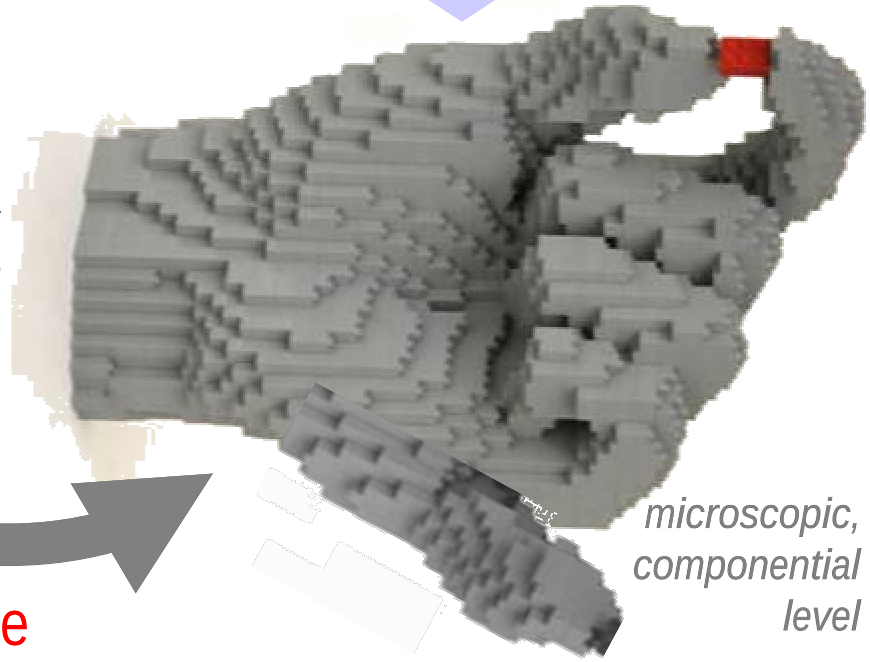
➢ Development: the missing link of the Modern Synthesis...

Amy L. Rawson
www.thirdroar.com

macroscopic, emergent level

“To understand novelty in evolution, we need to understand organisms down to their individual building blocks, down to their deepest components, for these are what undergo change.”



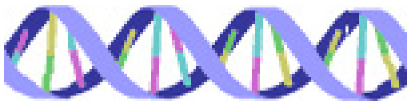


Nathan Sawaya
www.brickartist.com

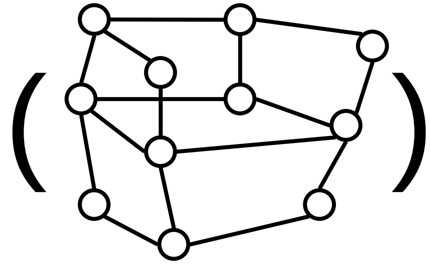
emergence

microscopic, componential level

- Development: the missing link of the Modern Synthesis...



more or less direct representation

≈  ≈

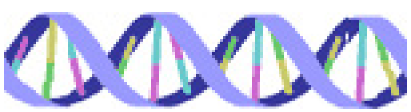
Amy L. Rawson
www.thirdroar.com

macroscopic, emergent level

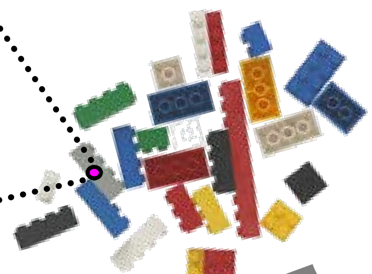
Genotype

"Transformation"?

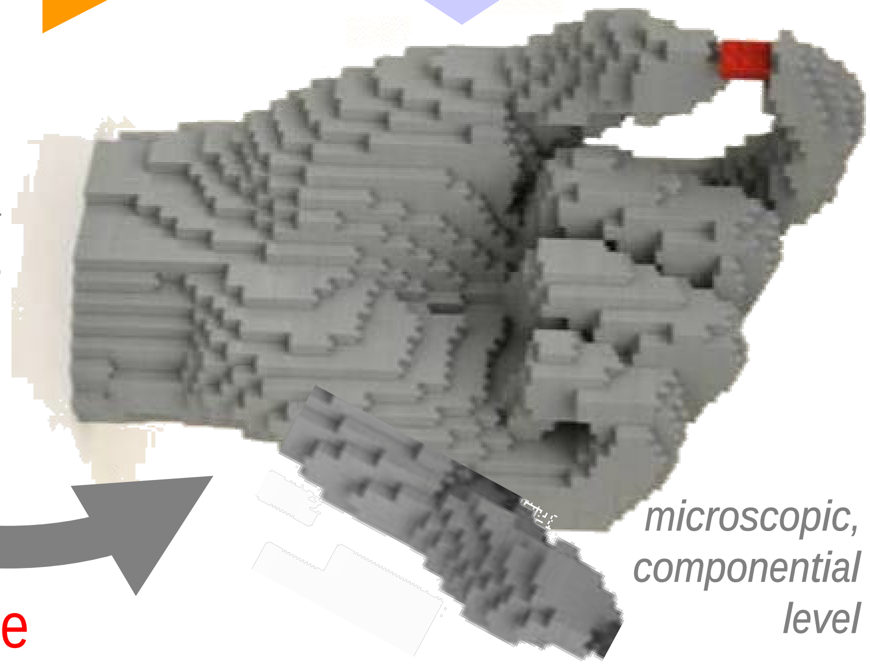
Phenotype



generic elementary rules of **self-assembly**



Nathan Sawaya
www.brickartist.com

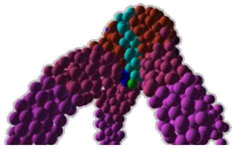


emergence

microscopic, componential level

The Promise of Unconventional / Natural Computing

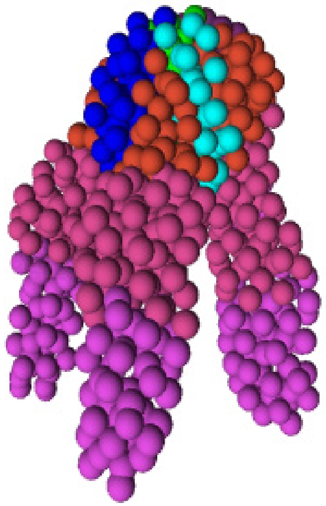
1. Tur(n)ing to Complex Systems (CS)
2. Engineering & Control of Self-Organization (ECSO)
3. Morphogenetic Engineering (ME)
   i.e. "Architectures without Architects"
4. Embryomorphic Engineering (EE)
   i.e. "Artificial Life Evo-Devo"

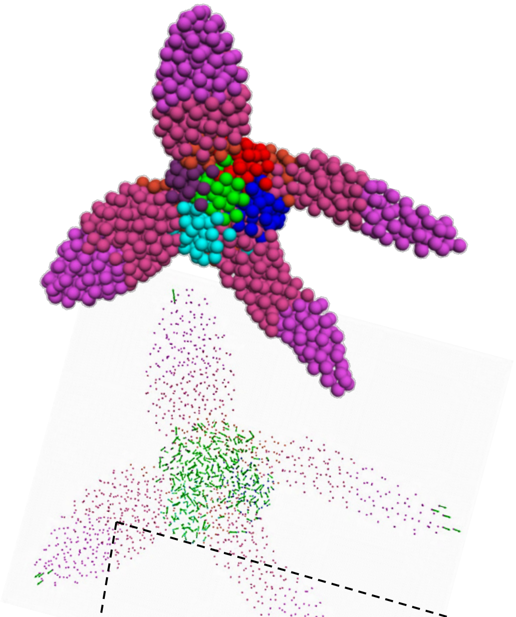


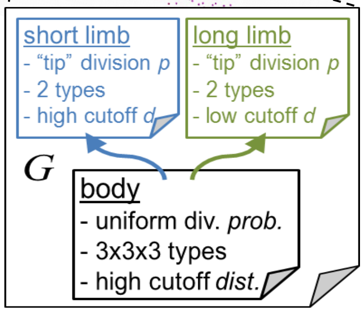
4.a. MapDevo
Modular Architecture by Programmable Development

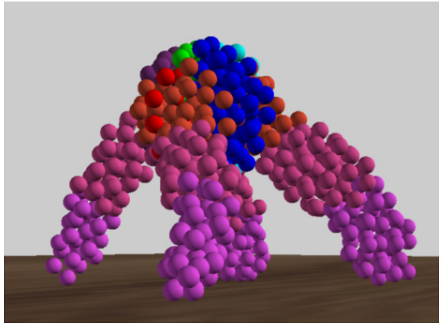
http://doursat.free.fr/mapdevo.html

➢ 3D Development





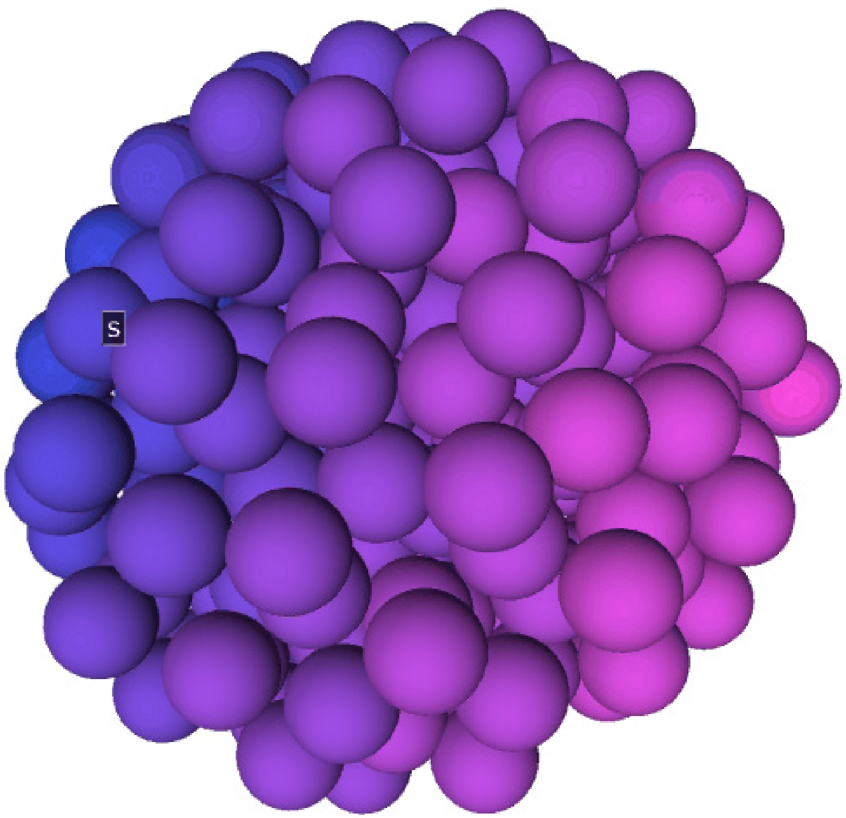


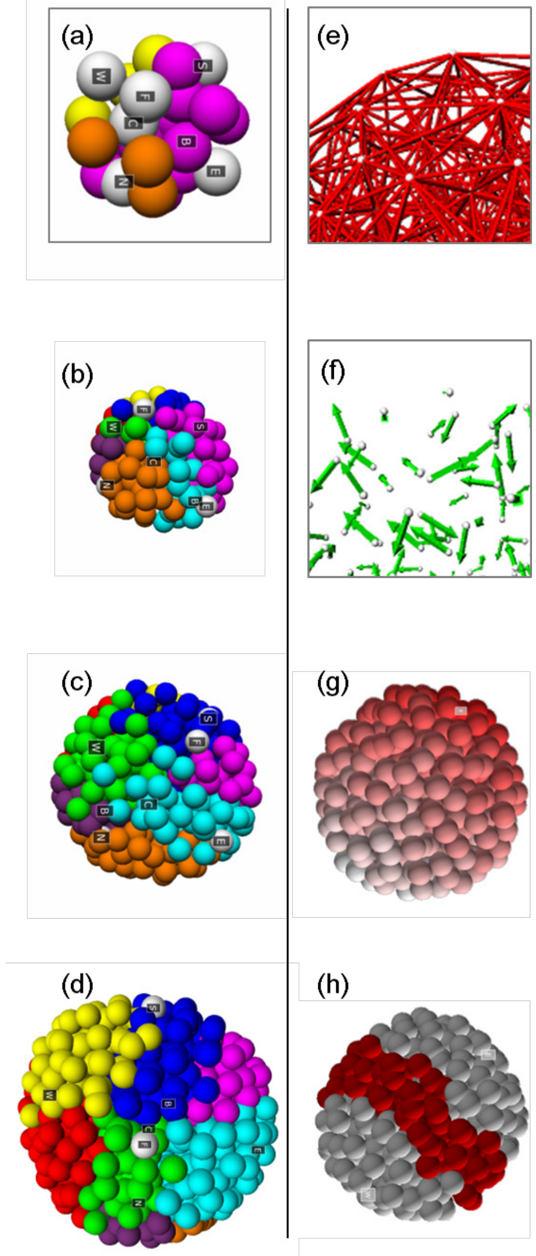


http://doursat.free.fr/mapdevo.html

Doursat, Sanchez, Fernandez, Kowaliw & Vico (2012)

➢ Body

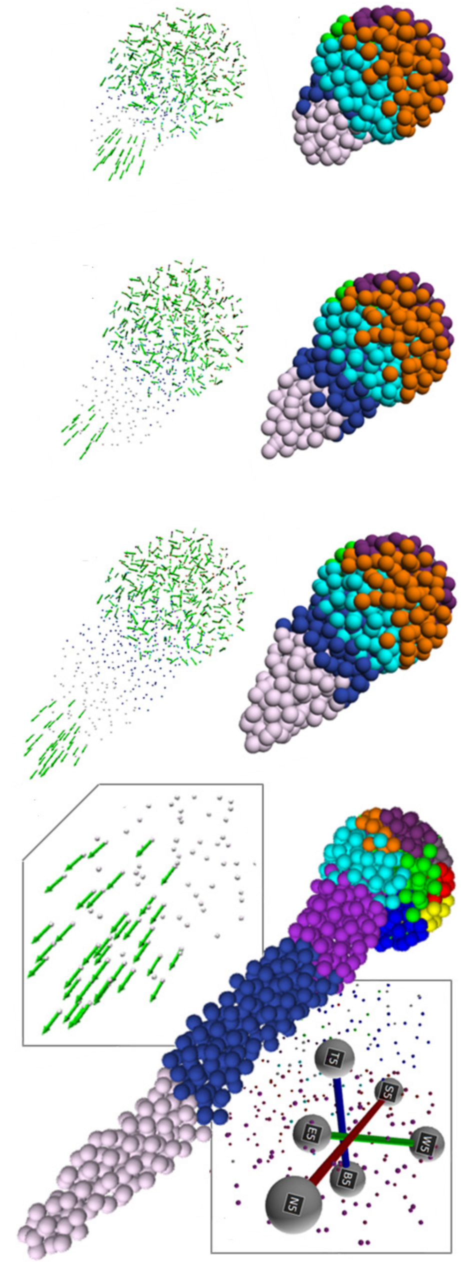




Doursat, Sanchez, Fernandez, Kowaliw & Vico (2012)

4.a. MapDevo – Modular Architecture

➢ Limbs

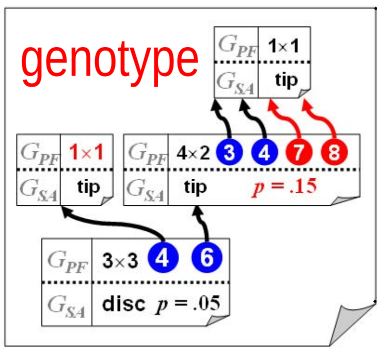


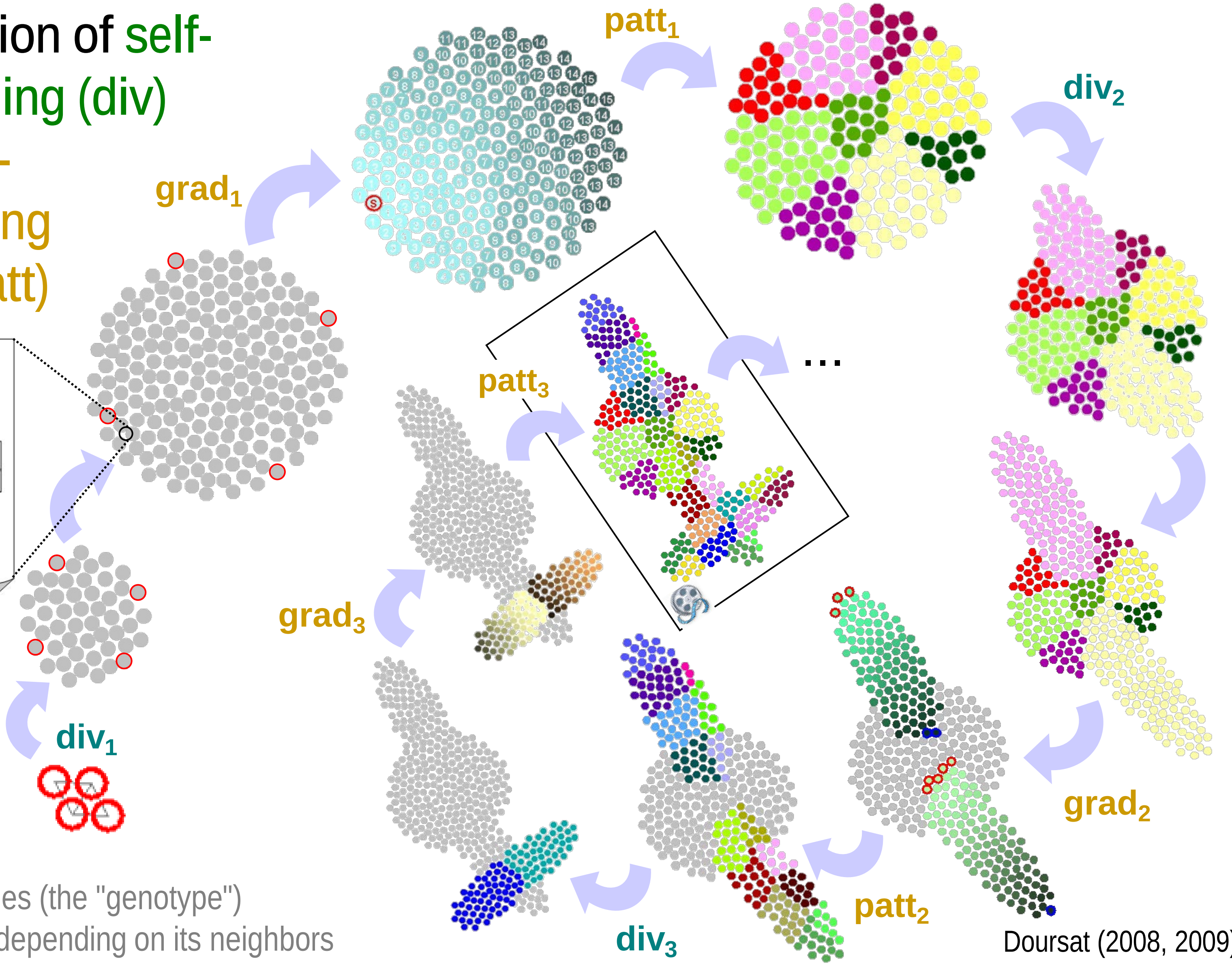
Doursat, Sanchez, Fernandez, Kowaliw & Vico (2012)

4.a. MapDevo – Modular Architecture

Capturing the essence of morphogenesis in an Artificial Life agent model

- Alternation of self-positioning (div) and self-identifying (grad/patt)

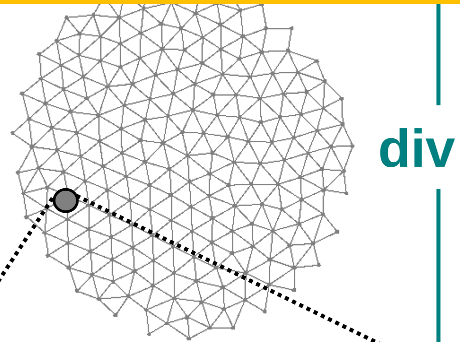




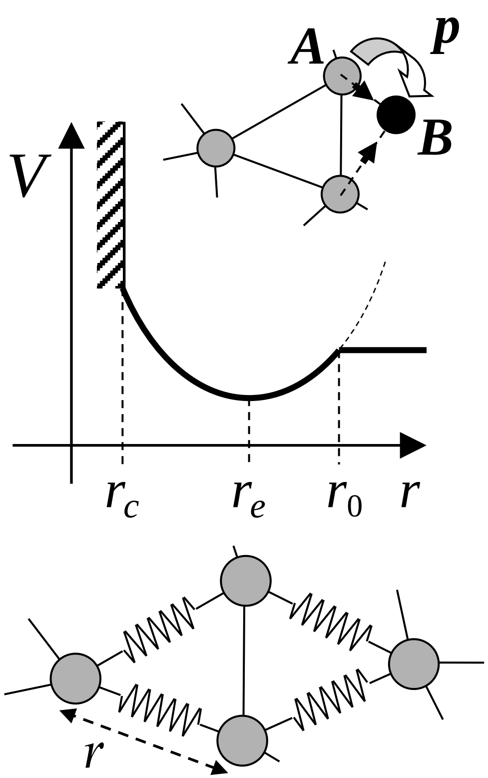
each agent
follows the same set
of self-architecting rules (the "genotype")
but reacts differently depending on its neighbors

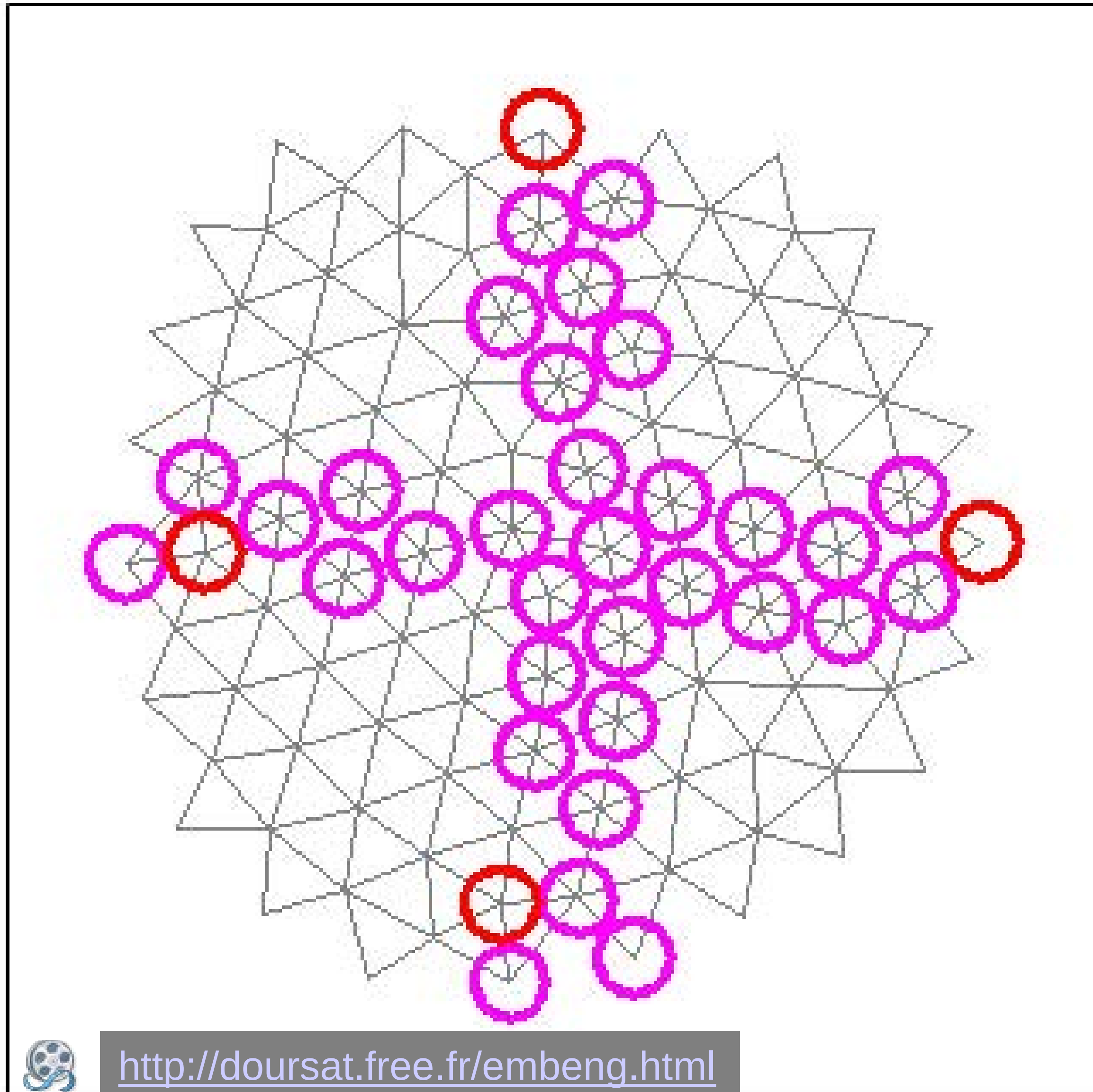
Doursat (2008, 2009)

4.a. MapDevo – Modular Architecture



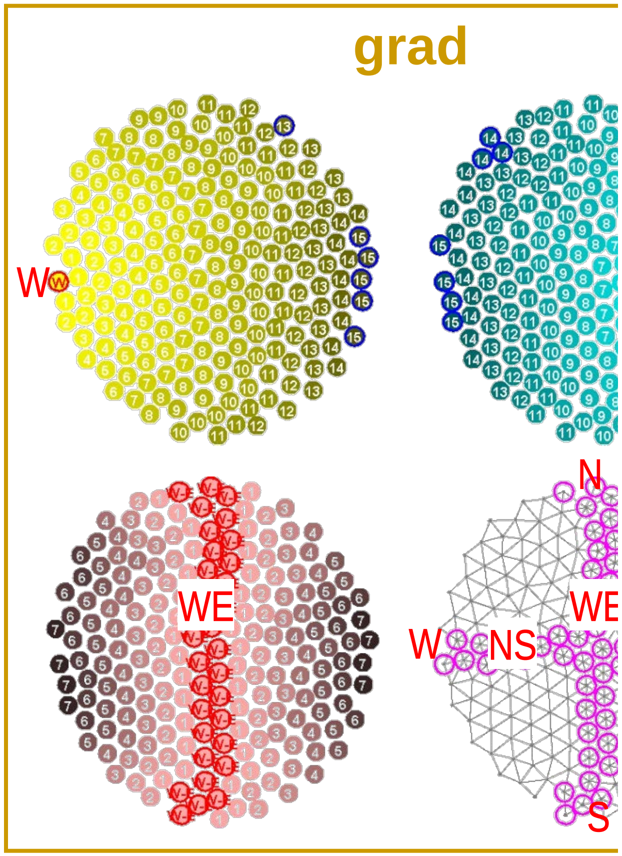
G_{SA}: $r_c < r_e = 1 << r_0$
$p = 0.05$

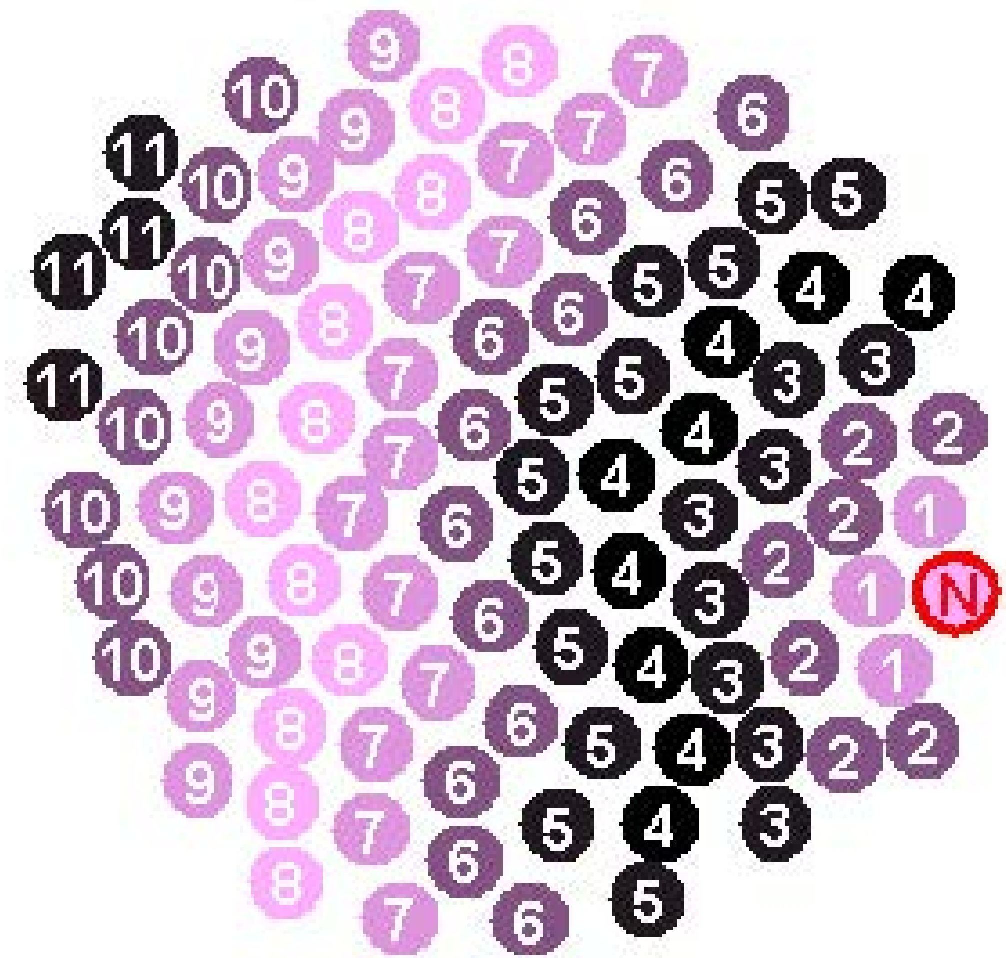




$$m\ddot{\mathbf{r}}^{cd} = -k\left(1 - \frac{r_0}{\left\|\mathbf{r}^{cd}\right\|}\right)\mathbf{r}^{cd} - \eta\dot{\mathbf{r}}^{cd}. \quad \Rightarrow \quad \Delta\mathbf{r}^{c} = -\Delta\mathbf{r}^{d} = \frac{\Delta\mathbf{r}^{cd}}{2} = -\frac{k}{2\eta}\left(1 - \frac{r_0}{\left\|\mathbf{r}^{cd}\right\|}\right)\mathbf{r}^{cd}.$$

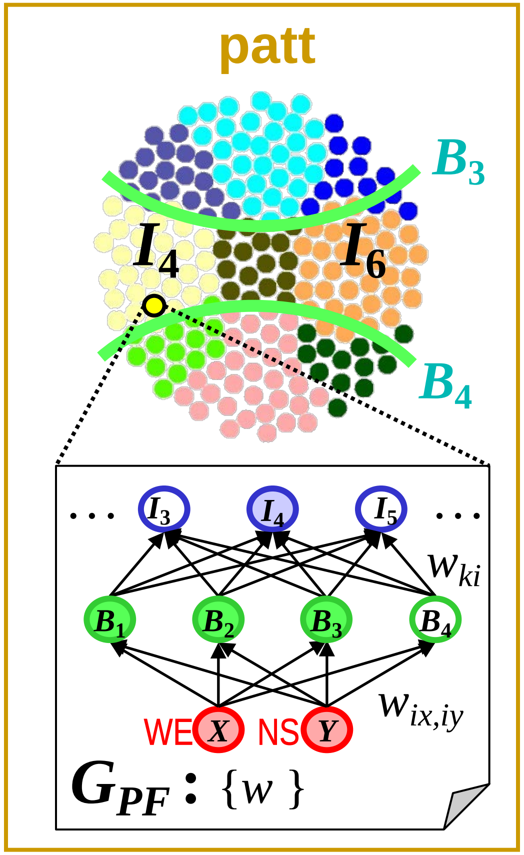
4.a. MapDevo – Modular Architecture

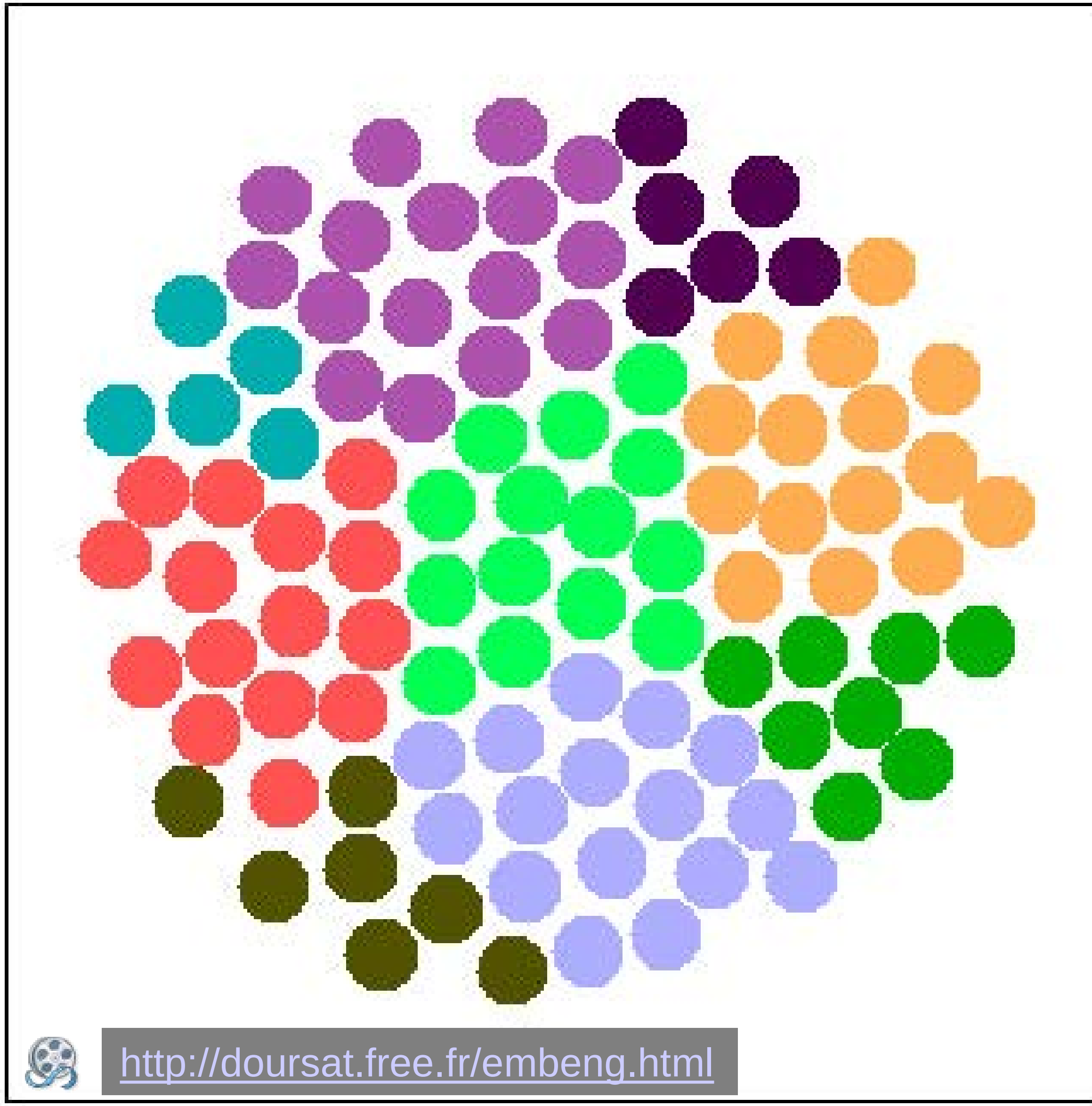




http://doursat.free.fr/embeng.html

4.a. MapDevo – Modular Architecture





$B_i = \sigma(L_i(X, Y)) = \sigma(w_{ix}\, X + w_{iy}\, Y - \theta_i)$

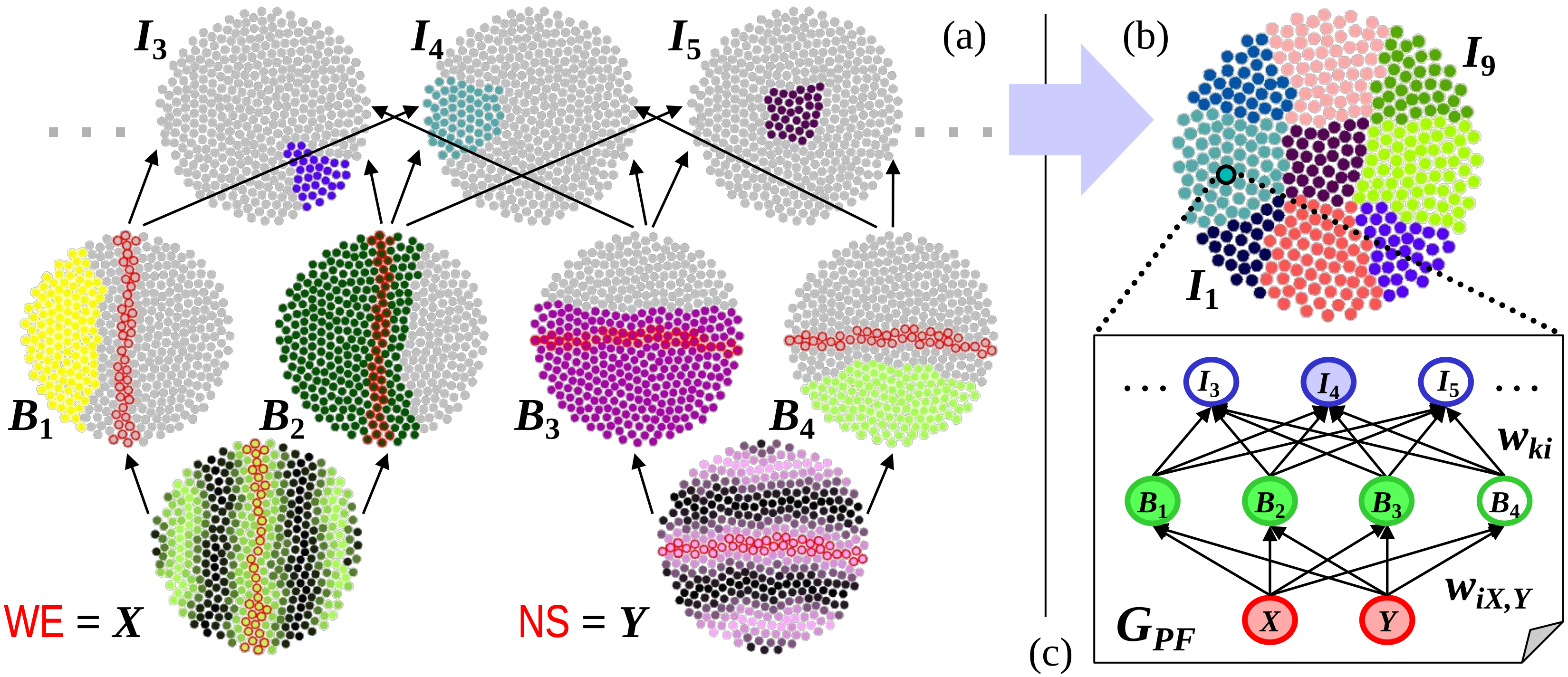
$I_k = \prod_i |w'_{ki}|(w'_{ki} B_i + (1 - w'_{ki})/2)$

➢ Programmed patterning (patt): the hidden embryo atlas

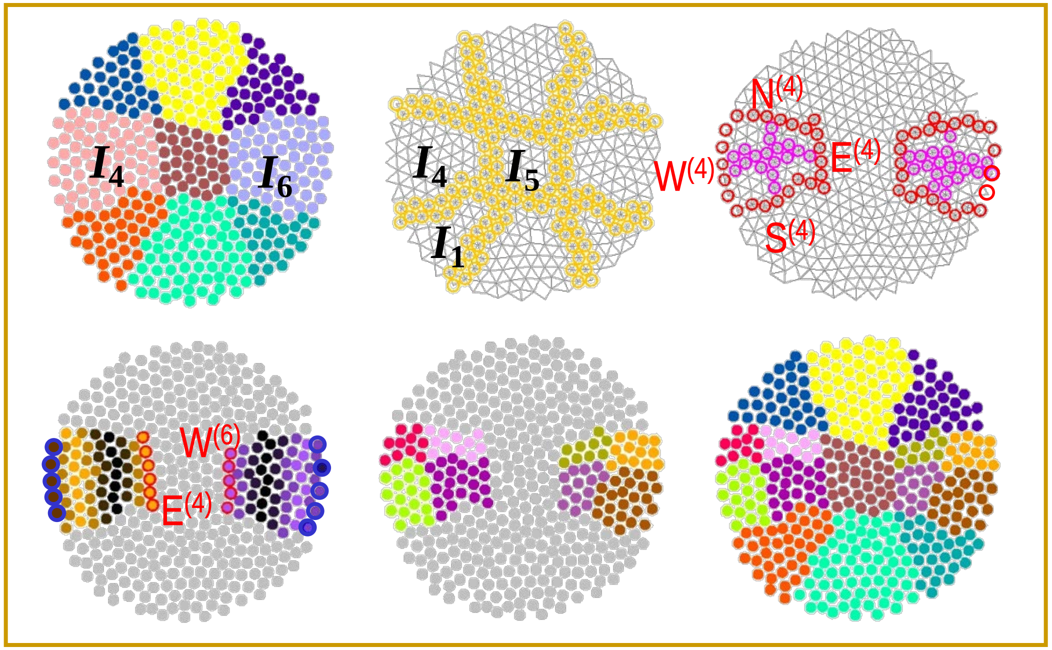
a) same swarm in different colormaps to visualize the agents' internal patterning variables X, Y, B_i and I_k (virtual *in situ hybridization)*

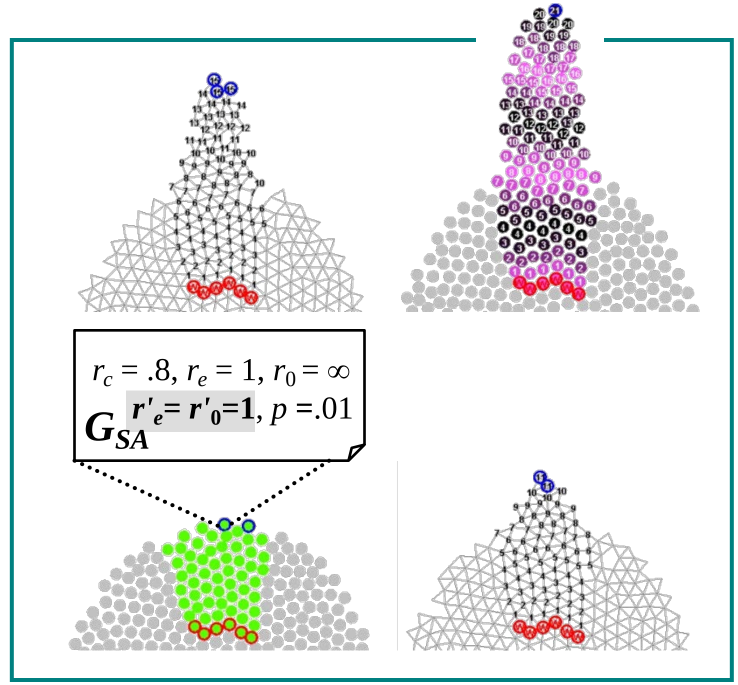
b) consolidated view of all identity regions I_k for $k = 1...9$

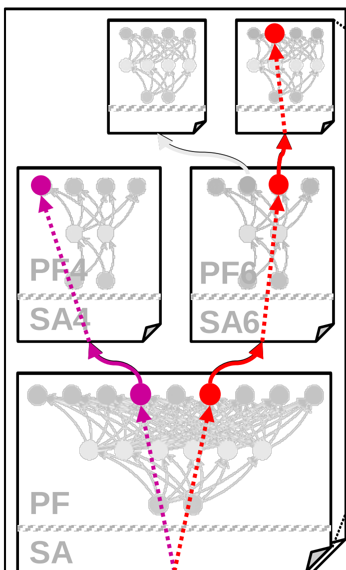
c) gene regulatory network used by each agent to calculate its expression levels, here: $B_1 = \sigma(1/3 - X)$, $B_3 = \sigma(2/3 - Y)$, $I_4 = B_1B_3(1 - B_4)$, etc.



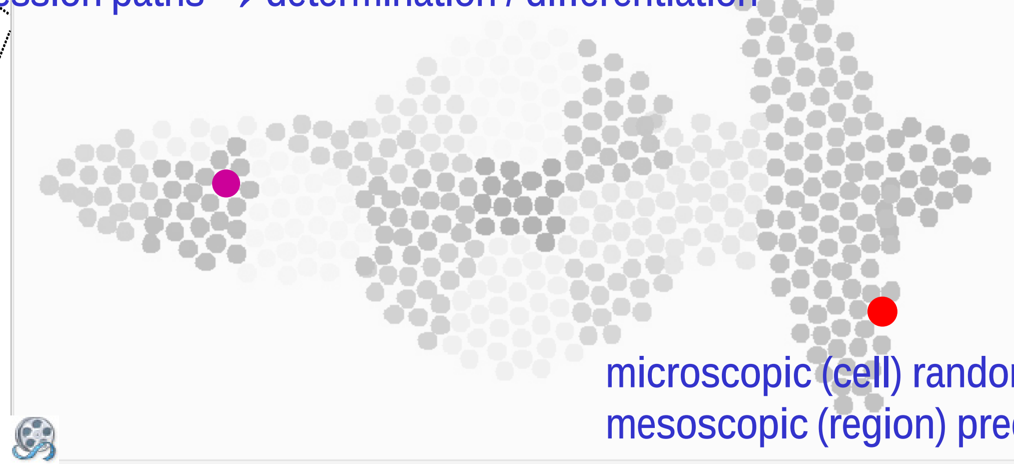
4.a. MapDevo – Modular Architecture







all cells have same GRN, but execute different expression paths → determination / differentiation

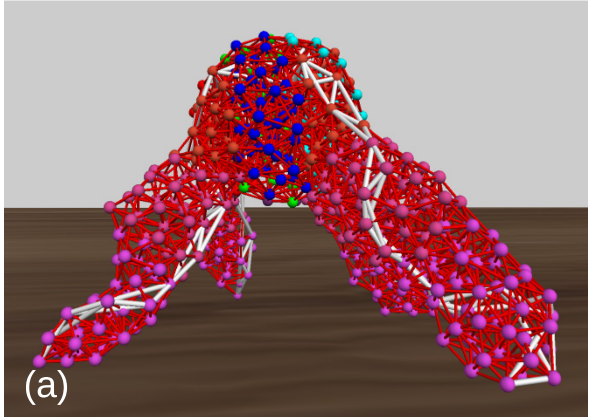


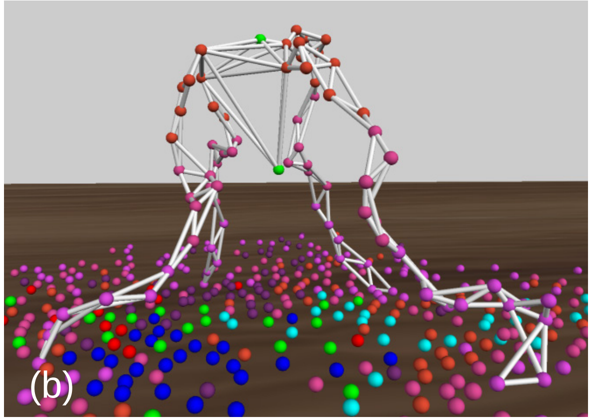
microscopic (cell) randomness, but mesoscopic (region) predictability

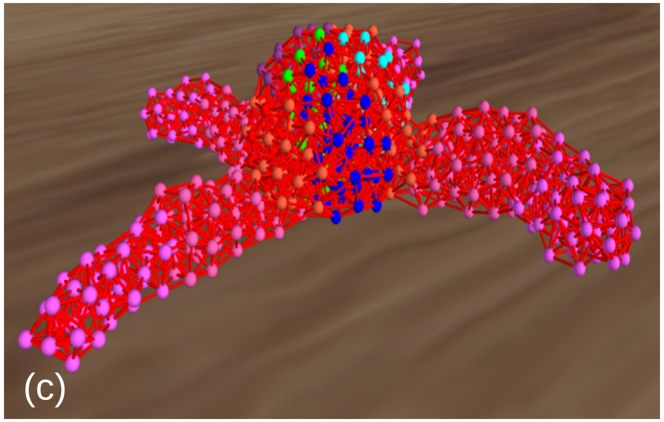
Doursat (2008, 2009)

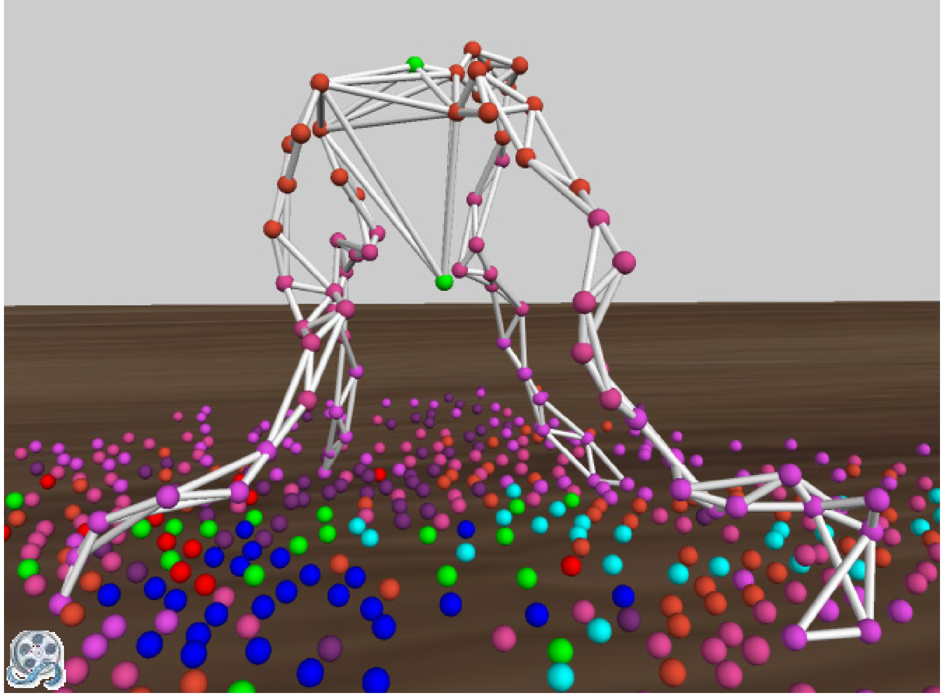
4.a. MapDevo – Modular Architecture

- Bones & muscles: structural differentiation and properties

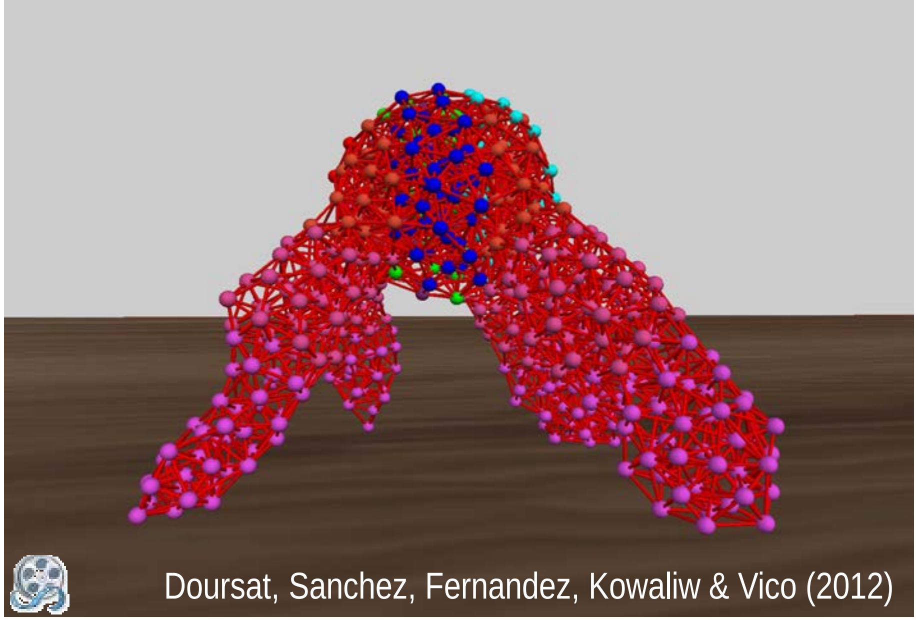








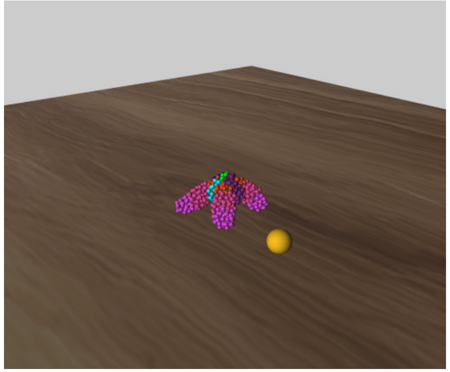
http://doursat.free.fr/mapdevo.html

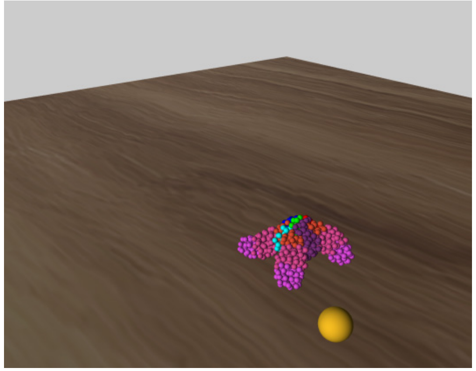


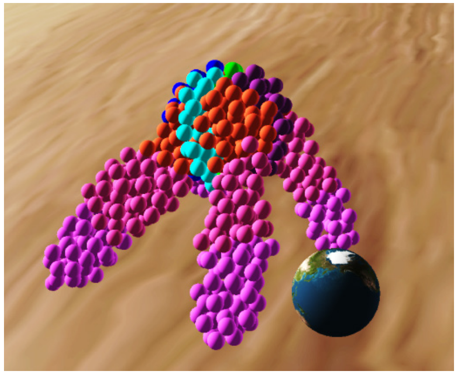
Doursat, Sanchez, Fernandez, Kowaliw & Vico (2012)

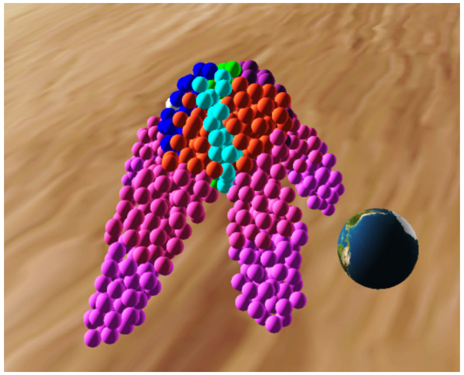
4.a. MapDevo – Modular Architecture

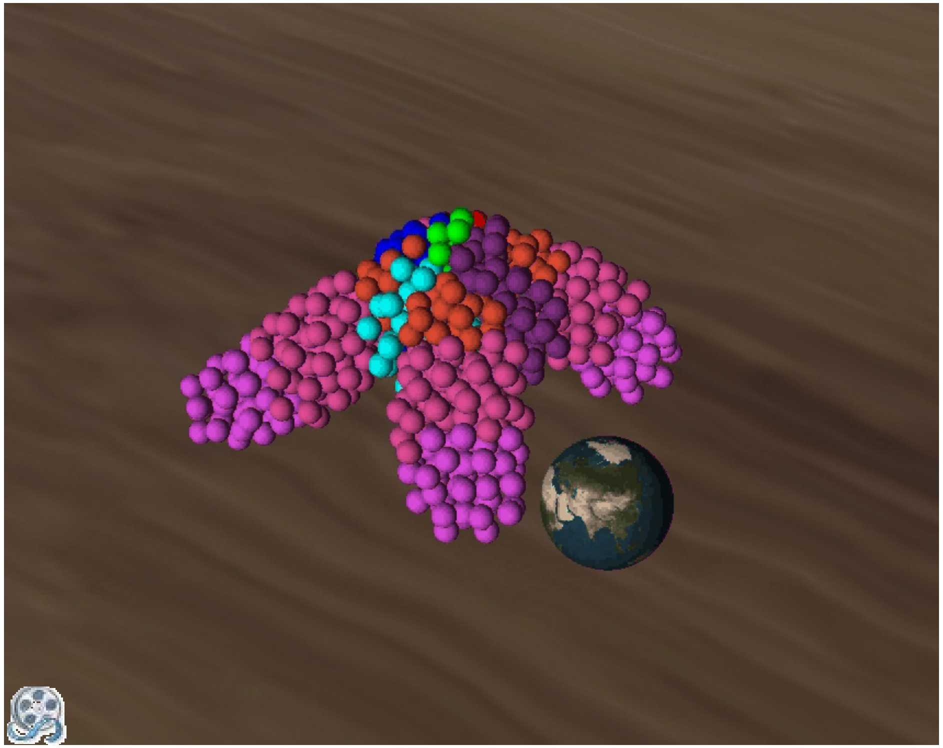
➢ Locomotion and behavior by muscle contraction

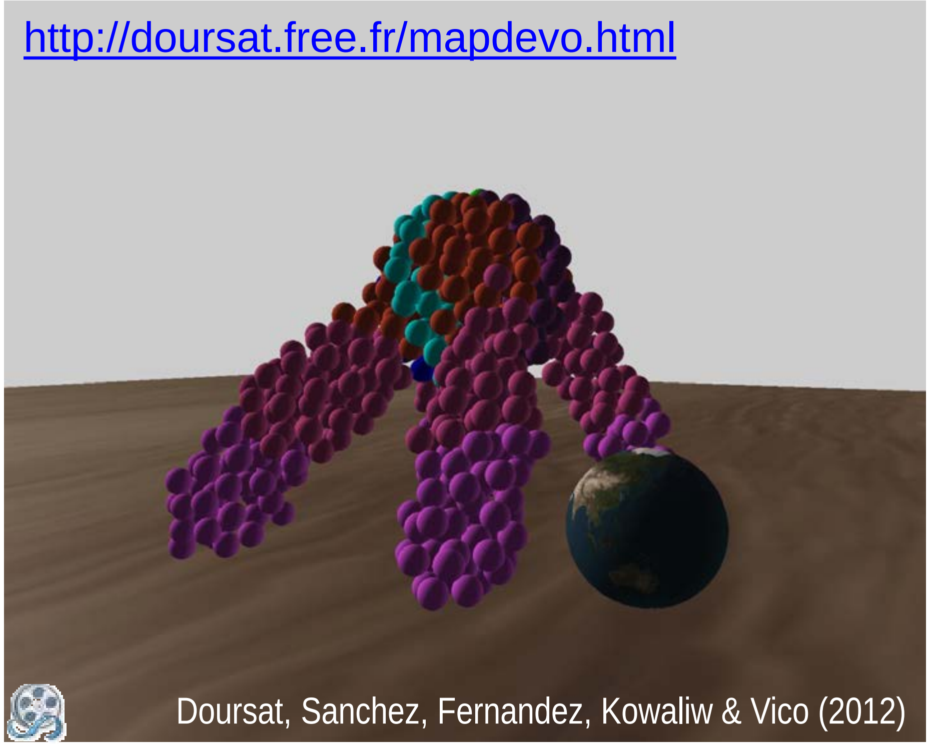




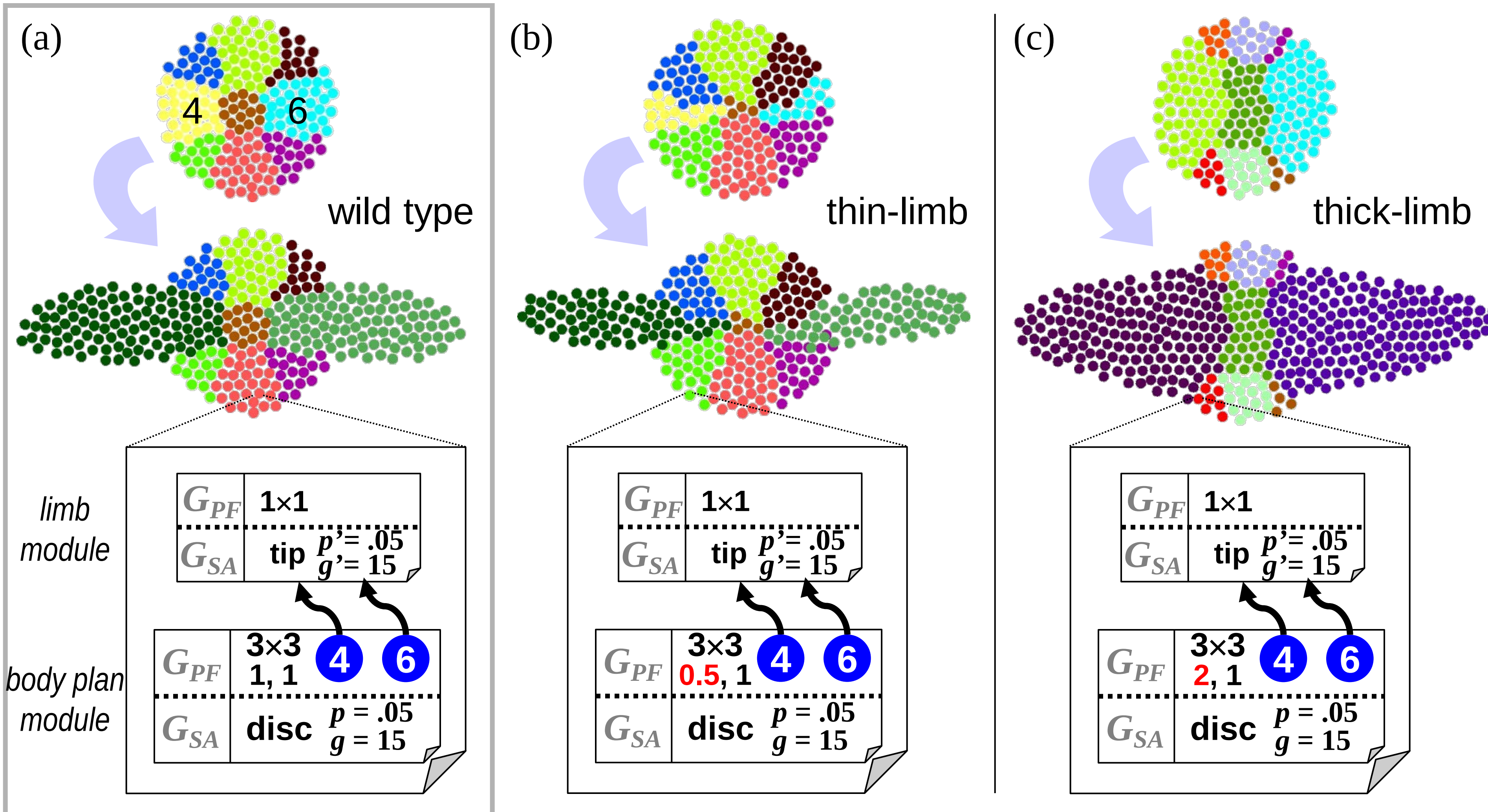








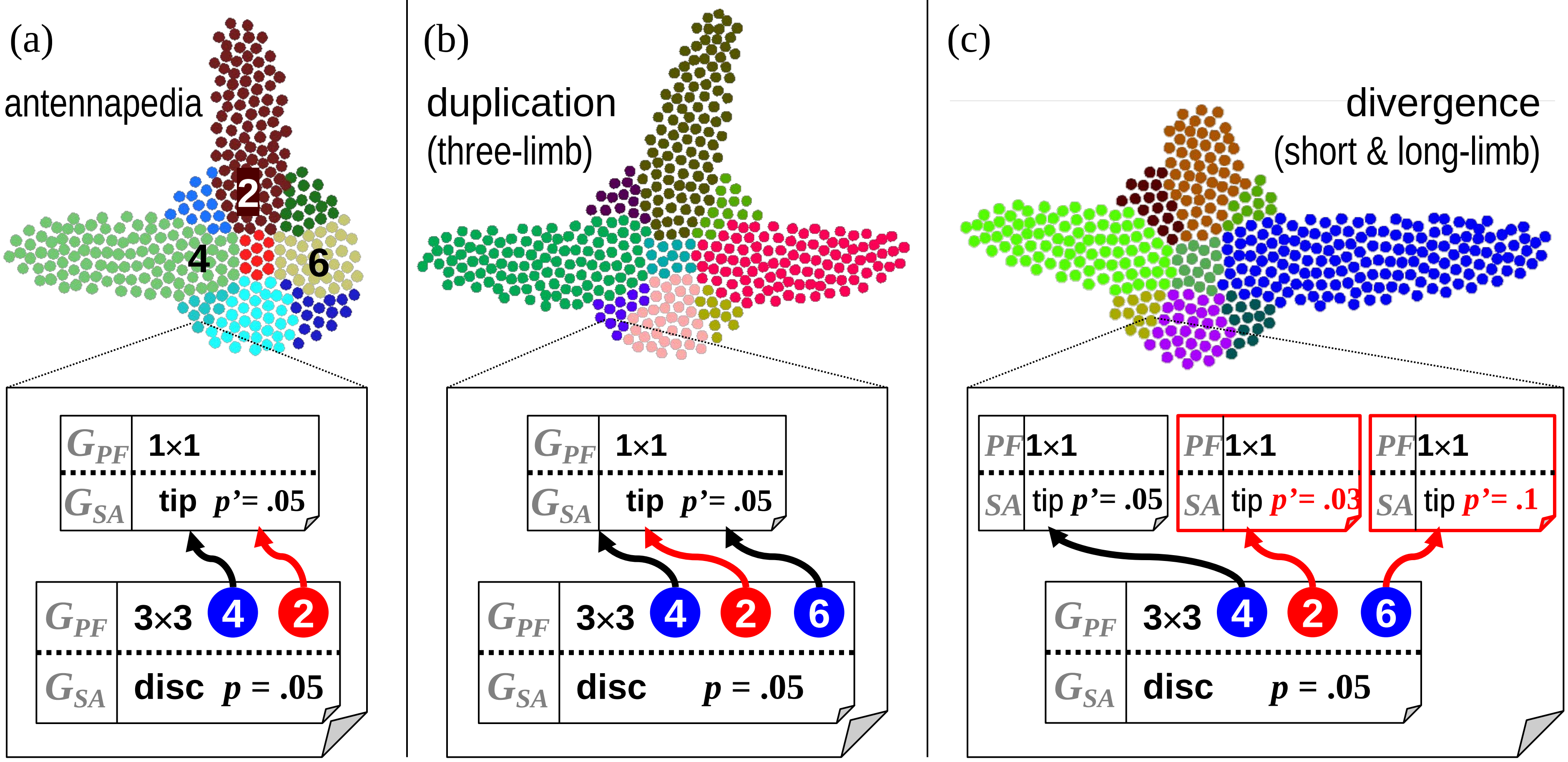
➢ Quantitative mutations: limb thickness



Doursat (2009)

- Qualitative mutations: limb position and differentiation

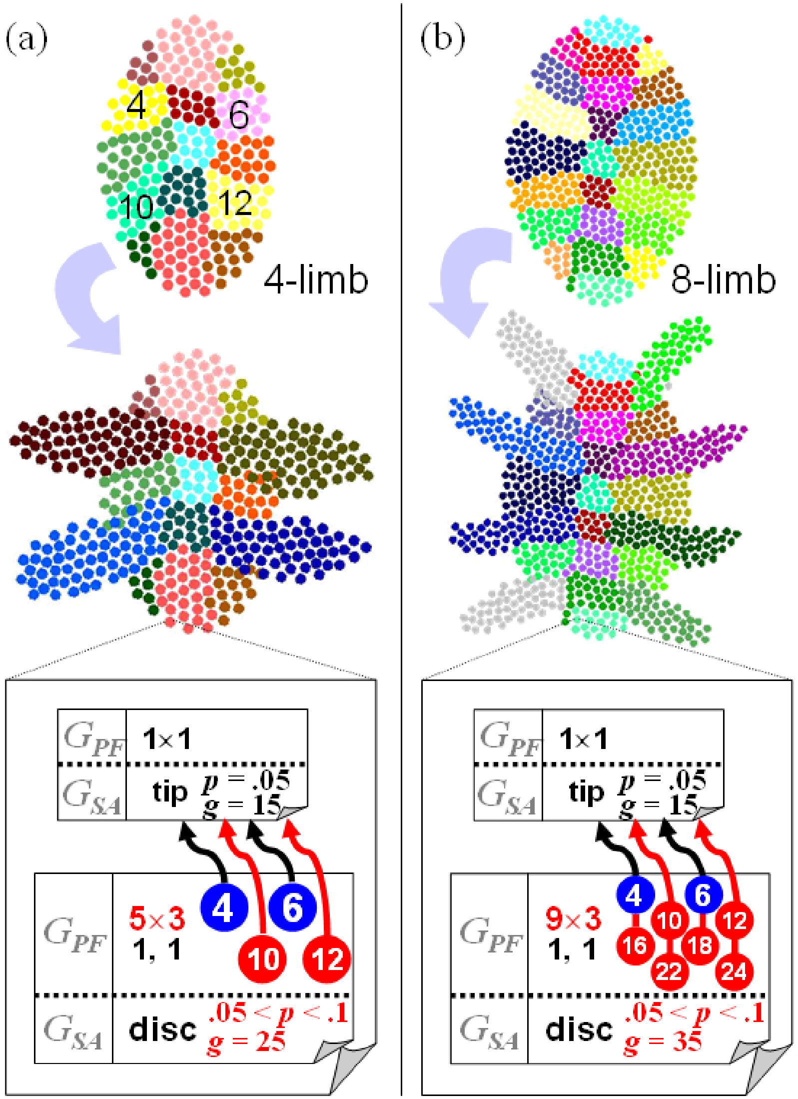
antennapedia | *homology* by duplication | divergence of the homology

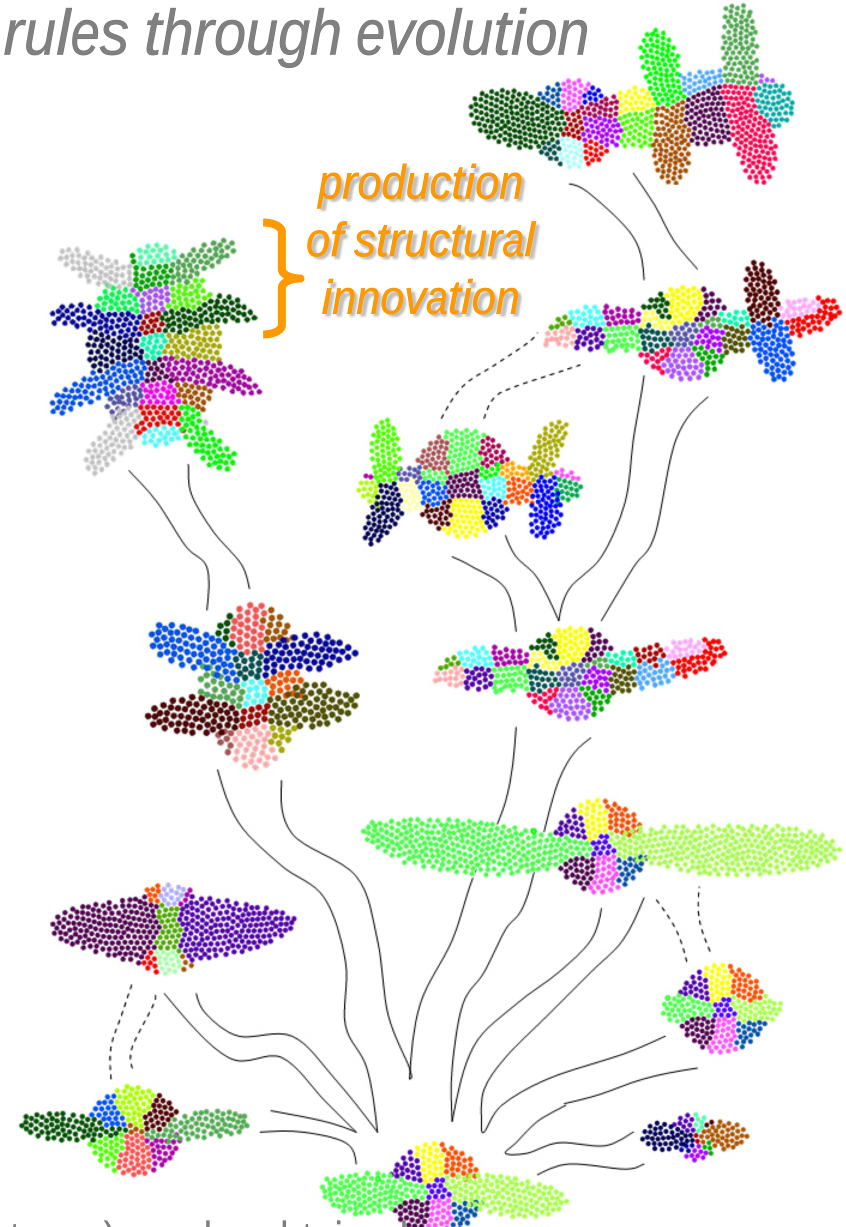


Doursat (2009)

4.a. MapDevo – Modular Architecture

Changing the agents' self-architecting rules through evolution





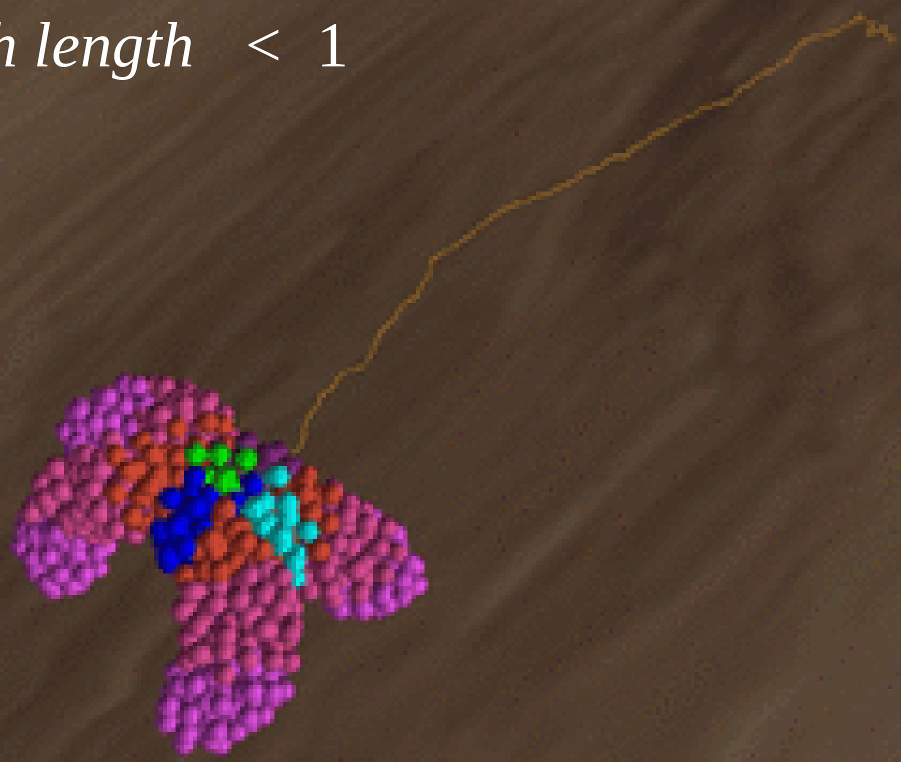
by tinkering with the genotype, new architectures (phenotypes) can be obtained

Doursat (2009)

4.a. MapDevo – Modular Architecture

- A simple challenge: path-based fitness

$$f = |\, end - start \,| \,/\, path\ length \;\; < \;\; 1$$

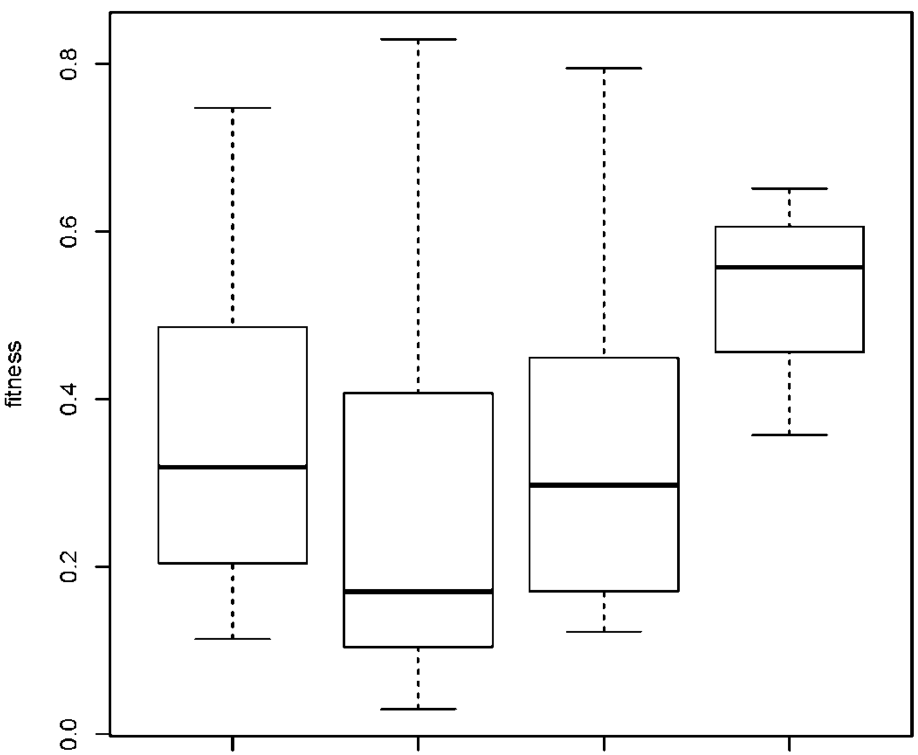


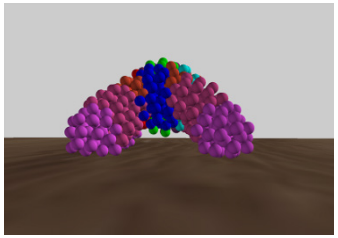
4.a. MapDevo – Modular Architecture

optimal body size?

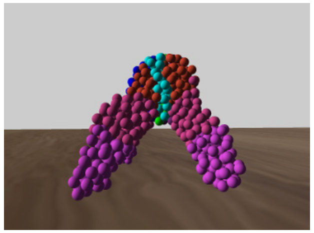
optimal limb size?

body size vs. fitness boxplots

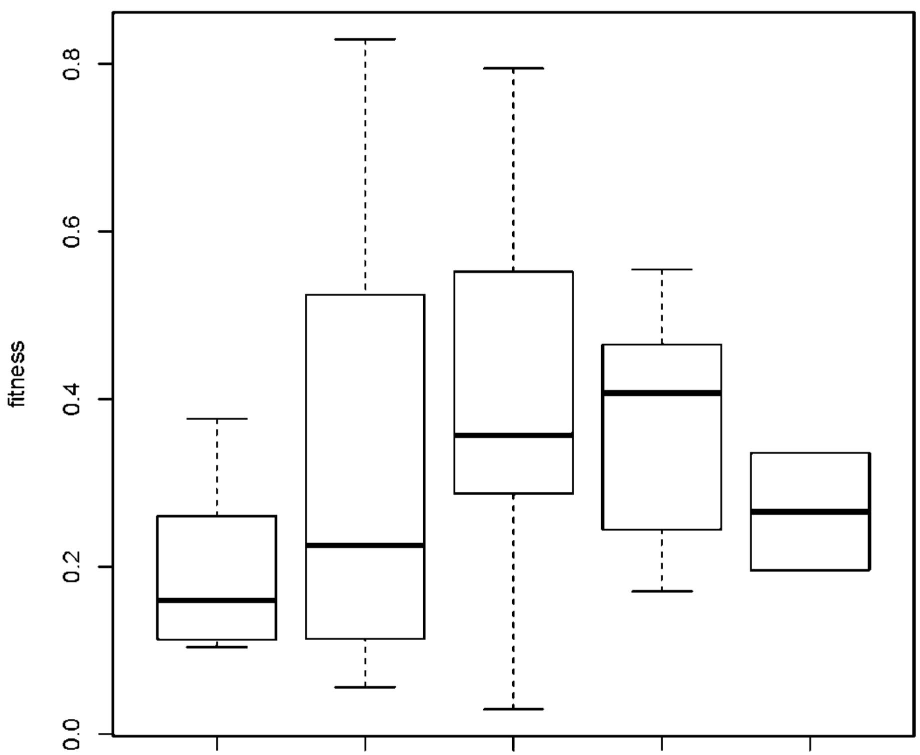




body size



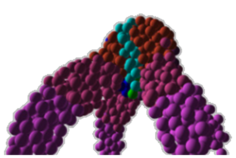
limb size vs. fitness boxplots



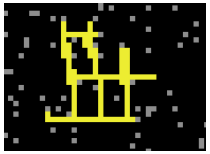
limb size

The Promise of Unconventional / Natural Computing

1. Tur(n)ing to Complex Systems (CS)
2. Engineering & Control of Self-Organization (ECSO)
3. Morphogenetic Engineering (ME)
   i.e. “Architectures without Architects”
4. Embryomorphic Engineering (EE)
   i.e. “Artificial Life Evo-Devo”

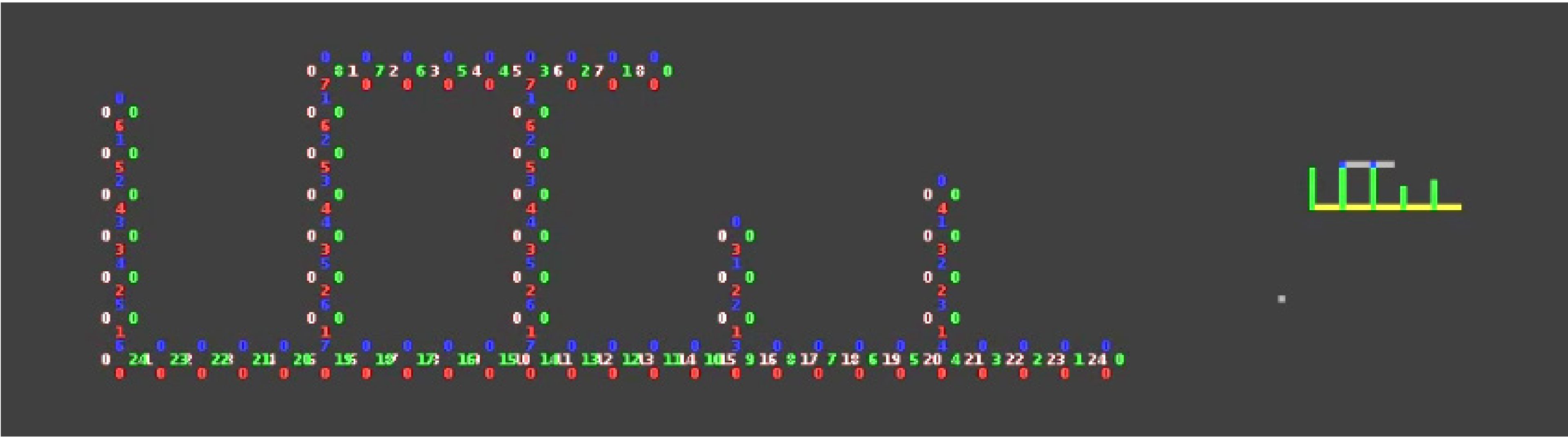


4.a. *MapDevo*
Modular Architecture by Programmable Development



4.b. ***ProgLim***
Program-Limited Aggregation

4.b. *ProgLim* – Program-Limited Aggregation

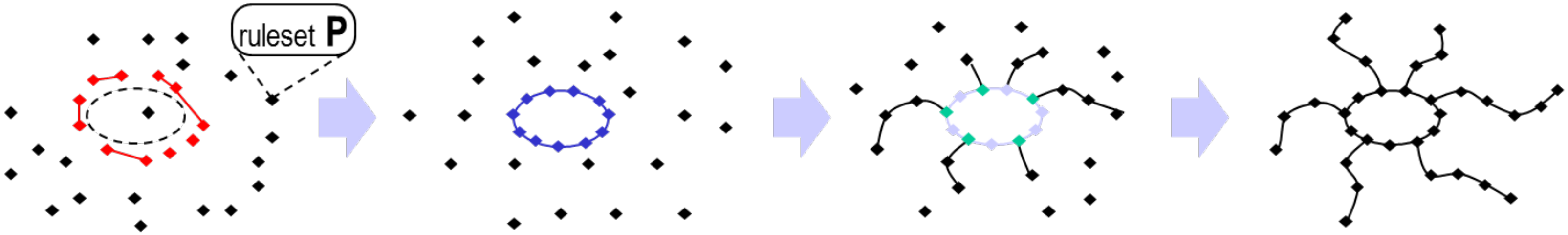


Doursat, Fourquet, Dordea & Kowaliw (2012)

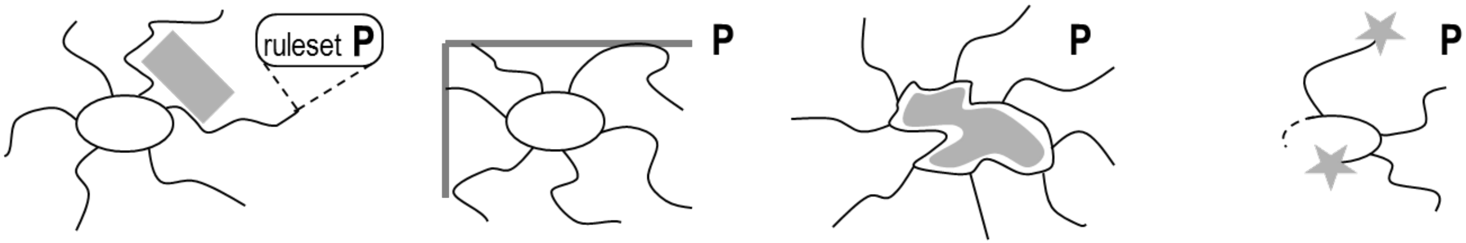
4.b. *ProgLim* – Program-Limited Aggregation

http://doursat.free.fr/nets.html

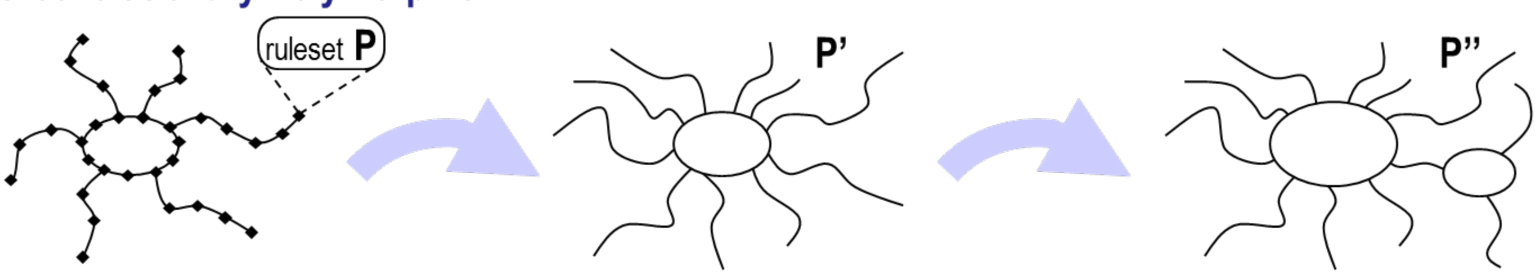
Stereotyped Development



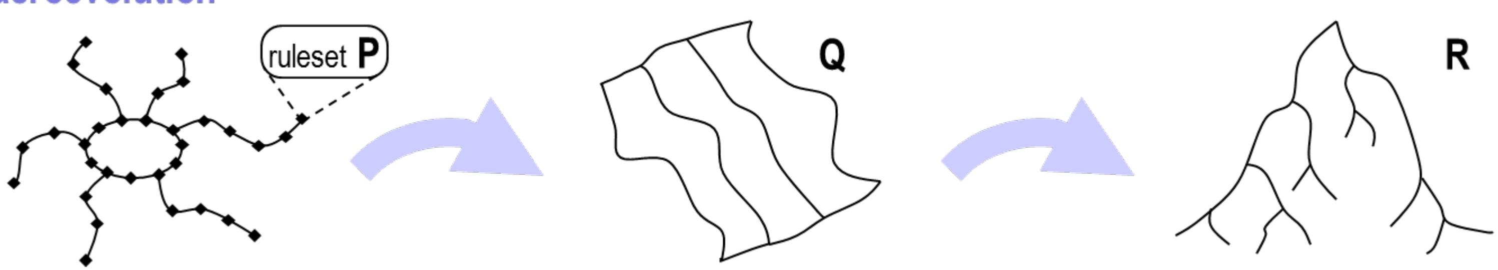
Environment-Induced Polyphenism



Microevolutionary Polymorphism

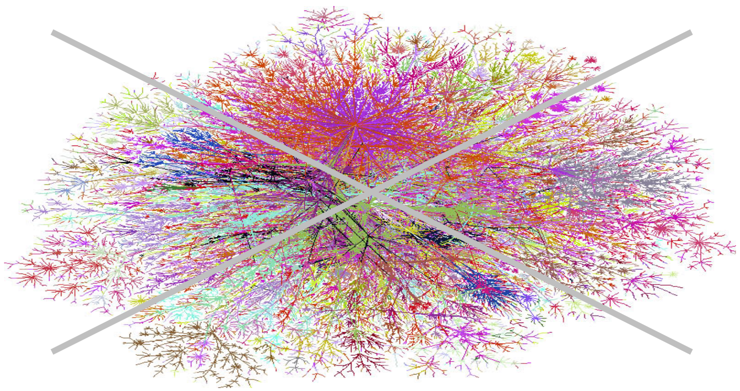


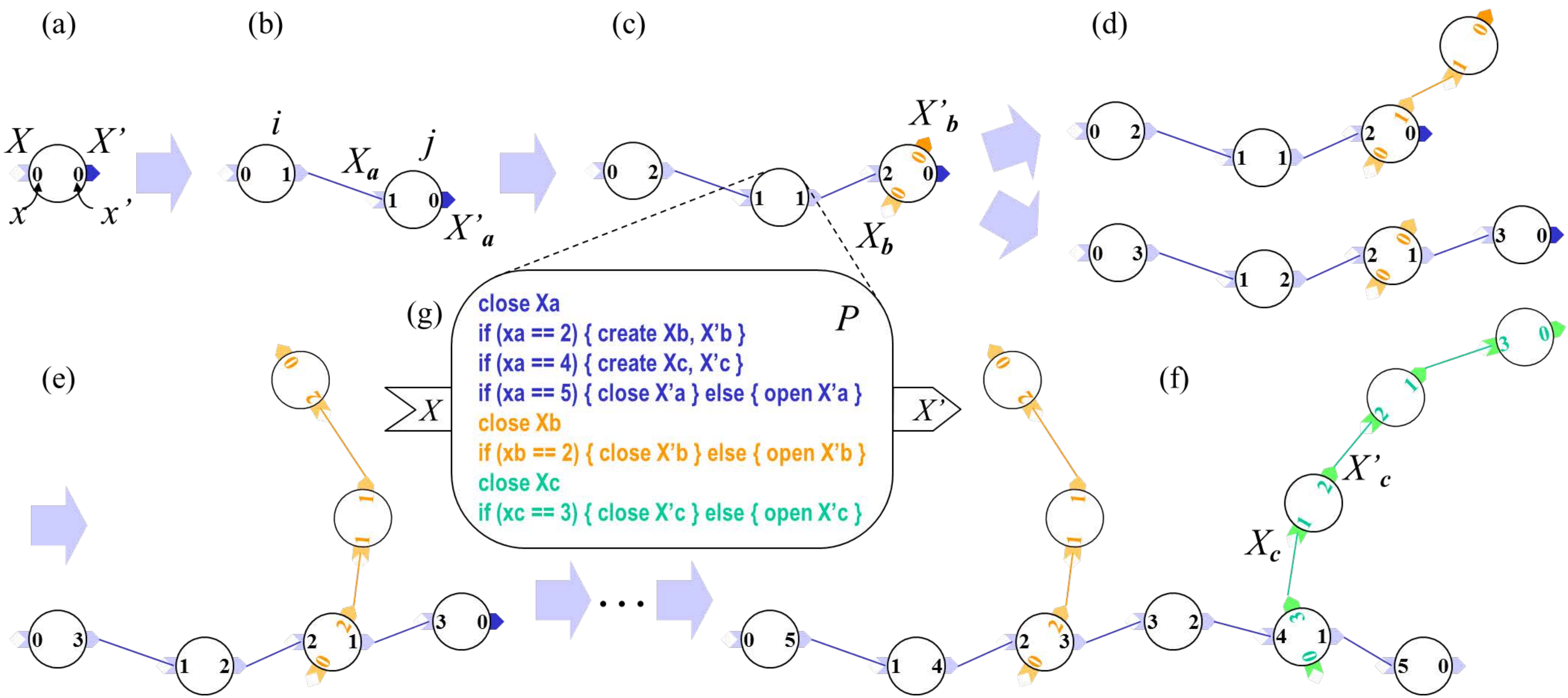
Macroevolution



4.b. *ProgLim* – Program-Limited Aggregation

- ~~Preferential (“scale-free”)~~
  Programmed
  Attachment
  Networking

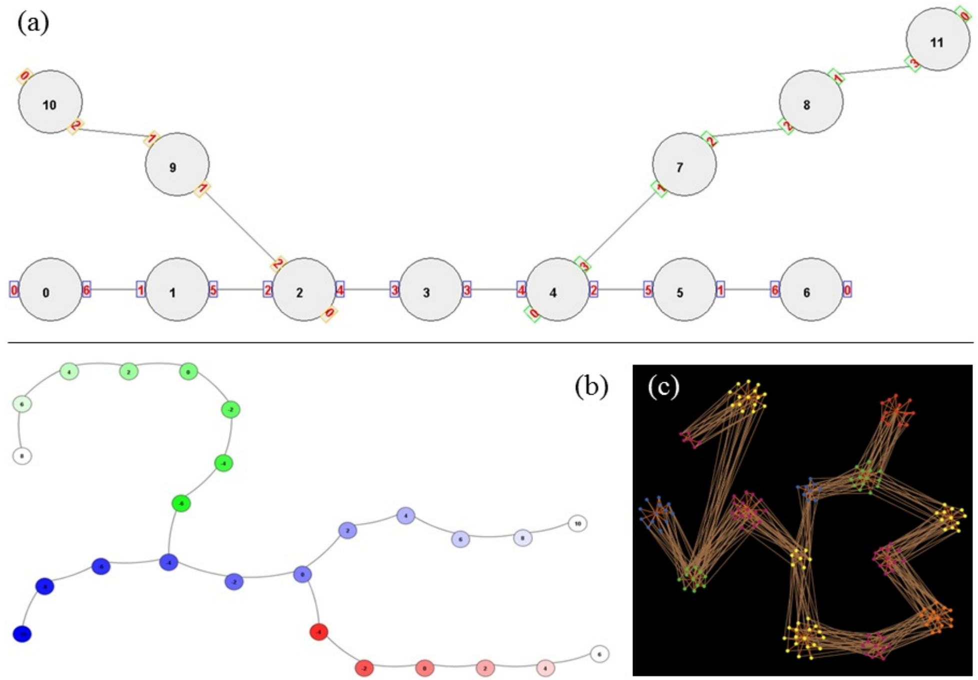




4.b. *ProgLim* – Program-Limited Aggregation

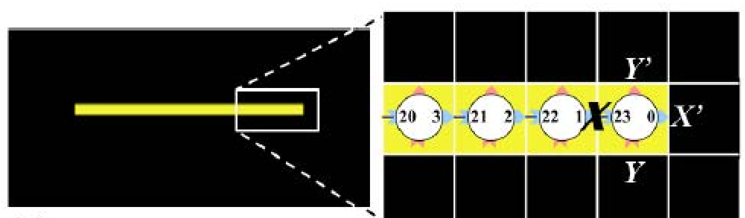
http://doursat.free.fr/prognet.html

- ~~Preferential~~
  Programmed Attachment Networking
  (ProgNet)



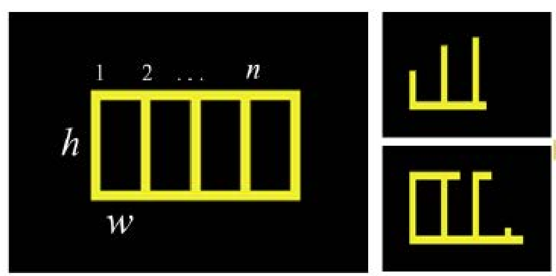
http://doursat.free.fr/proglim.html

- ~~Diffusion-~~
  Program-Limited Aggregation
  (ProgLim)



(a)

Port Routine $P1$: Line
open X, X'
close Y, Y'



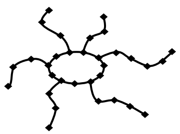
(b)

Port Routine $P2$: Row of Rectangles
$h = 10, w = 5, n = 4$
if $(x' = 0)$ then open X'
if $(x \,\%\, w = 0)$ then open Y'
if $(y > 0)$ then close X'
if $(y = h)$ then open X'
if $(x \geq n.w)$ then close X'
if $(y \geq h)$ then close Y'

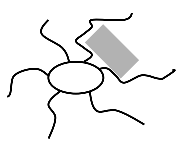
Doursat, Fourquet, Dordea & Kowaliw (2012)

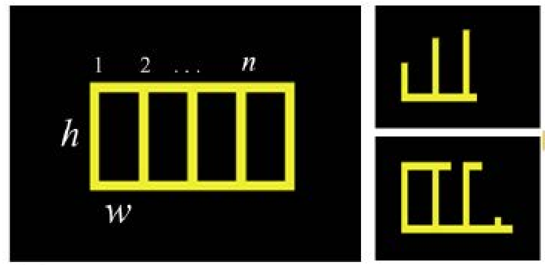
4.b. *ProgLim* – Program-Limited Aggregation

- Stereotyped development



- Environment-Induced Polyphenism

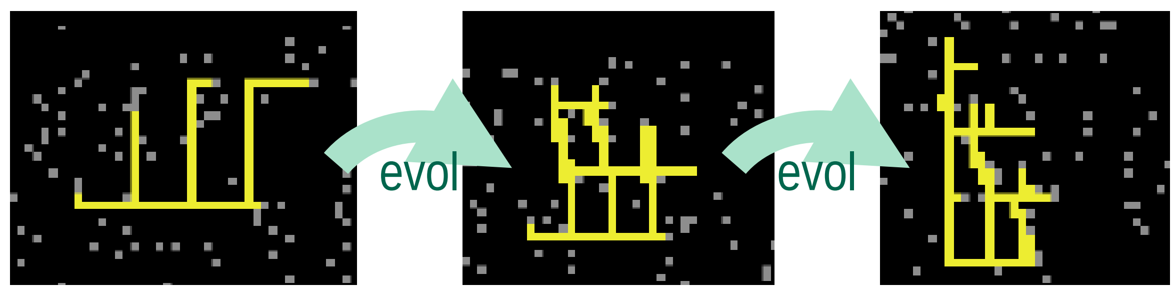




(b)

Port Routine $P2$: Row of Rectangles

$h = 10, w = 5, n = 4$
if $(x' = 0)$ then open X'
if $(x \% w = 0)$ then open Y'
if $(y > 0)$ then close X'
if $(y = h)$ then open X'
if $(x \geq n.w)$ then close X'
if $(y \geq h)$ then close Y'



(a) (b) (c)

Port Routine $P5$: Rectangles (b)

$h = 8, w = 5, n = 4, c = 9$
if $(y \% h = 0)$ then open X'
if $(x \% w = 0)$ then open Y'
if $(y = h)$ then close Y'
if $(x = n.w)$ then close X'
if $(r(Y') = 1)$ then open X
if $(y' > c \ \& \ r(X') = 1)$ then open X

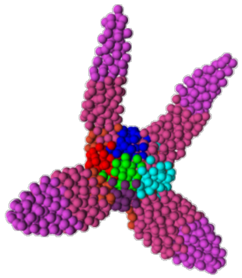
Port Routine $P6$: Rectangles (c)

$h = 8, w = 5$
if $(y \% h = 0)$ then open X'
if $(x \% w = 0)$ then open Y'
if $(r(X) = 1)$ then open Y'
if $(r(X') = 1)$ then open X

Doursat, Fourquet, Dordea & Kowaliw (2012)

Acknowledgments

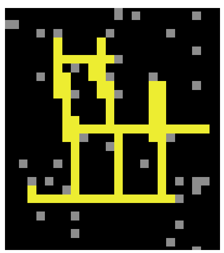
Carlos A. Sánchez



Universidad de Málaga, Spain

grupo de estudios en biomimética

David Fourquet



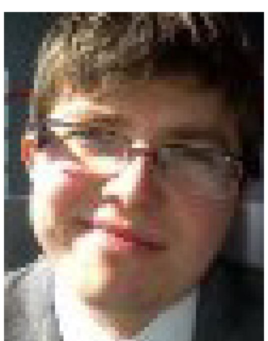
Ecole Polytechnique, Paris, France

INSTITUT DES SYSTEMES COMPLEXES

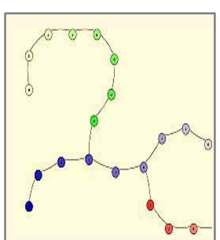
Taras Kowaliw

Ecole Polytechnique, Paris, France

INSTITUT DES SYSTEMES COMPLEXES



Razvan Dordea



Ecole Polytechnique, Paris, France

INSTITUT DES SYSTEMES COMPLEXES