BIONETICS 2007
Bio-Inspired Models of Network, Information, and Computing Systems
Budapest, December 10-13, 2007

How to plan self-organization, control decentralization, and design evolution:

Addressing the paradoxes of complex systems engineering with metaphors from biological development

René Doursat

Institut des Systèmes Complexes, CREA
CNRS & Ecole Polytechnique
57-59, rue Lhomond, 75005 Paris, France

Complex Systems Engineering from Biological Development

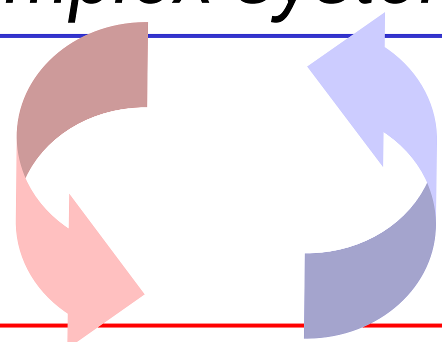
1. The Meta-Design of Complexity
2. Concrete Models Based on Multicellular Systems
   a. Embryomorphic Engineering
   b. Neurodynamic Pattern Recognition
   c. Immune Network Security
3. Planning for Autonomy

1. The Meta-Design of Complexity

➢ Complex systems engineering

understanding natural complex systems

exporting:
- decentralization
- autonomy
- evolution

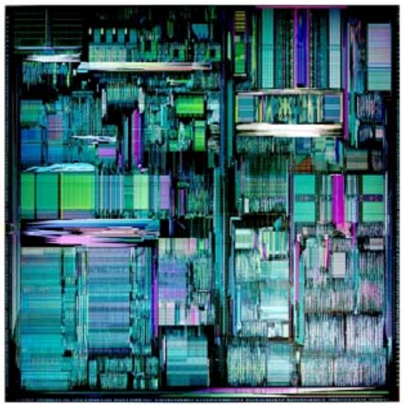


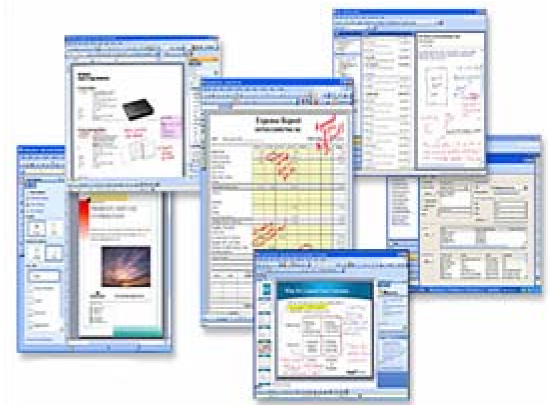
importing:
- modeling
- simulation

designing a new generation of artificial systems

1. The Meta-Design of Complexity

➢ Exploding size & complexity of computer systems

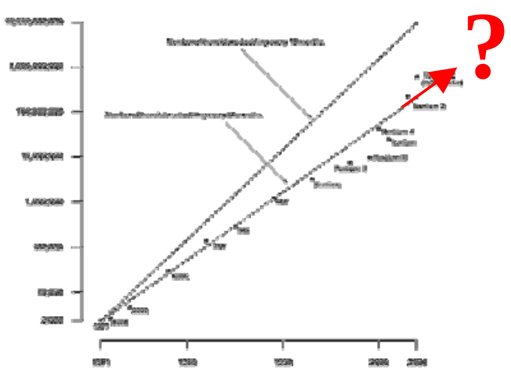




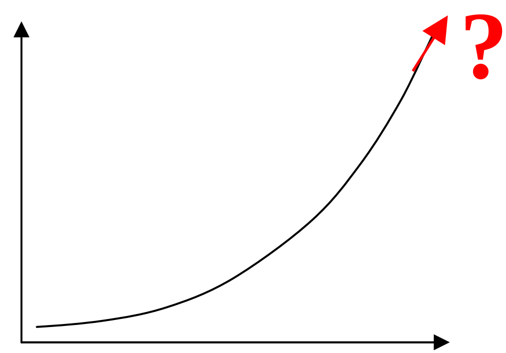
whether hardware,
in the number and interconnection
of integrated components,

software,
in the number and interconnection
of functions, modules, and layers,

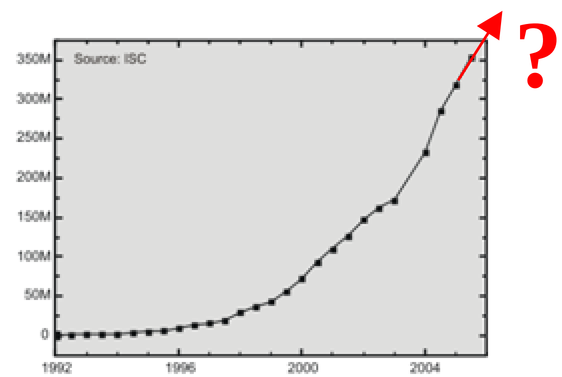
or networks, ...
in the number and interconnection
of distributed applications (C/S) and users,



number of transistors/year



number of O/S lines of code/year



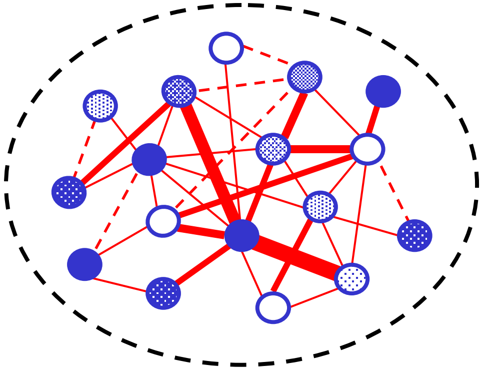
number of network hosts/year

1. The Meta-Design of Complexity

- ... leads us toward decentralized, automous systems...
  - ✓ information architects and engineers are already beginning to lose grip on their creation, which exceeds the capacity of a single human mind
  - ✓ there is a *de facto* segmentation and distribution of the traditionally centralized control over systems design
  - ✓ the march toward decentralization has already begun: larger teams of engineers, open source communities, collaborative work around “modules”
  - ✓ thus, rather than insisting on rigidly designing every detail, the trend should be to “step back” even further: focus on establishing the **generic conditions** that will allow systems to self-assemble, self-regulate & evolve
  - ✓ in fact, future progress in ICT could ultimately depend on our ability to foster systems that **endogenously**
    - grow, function, and repair themselves
    - adapt and improve

1. The Meta-Design of Complexity

➢ ... and rethink engineering in terms of *complex systems*



- (very) **large** number of (light-weight) elements interact **locally**
- **simple** individual rules create a complex **emergent** behavior
- **decentralized** dynamics: no master blueprint or architect
- **self-organization** and **evolution** of innovative order

✓ in particular, seek inspiration from biological and social systems

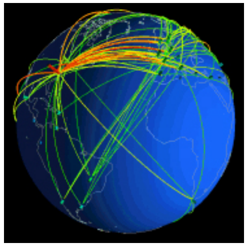
physical pattern formation

organism development

insect colonies

the brain

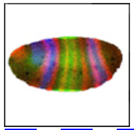
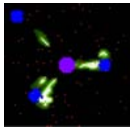
World Wide Web

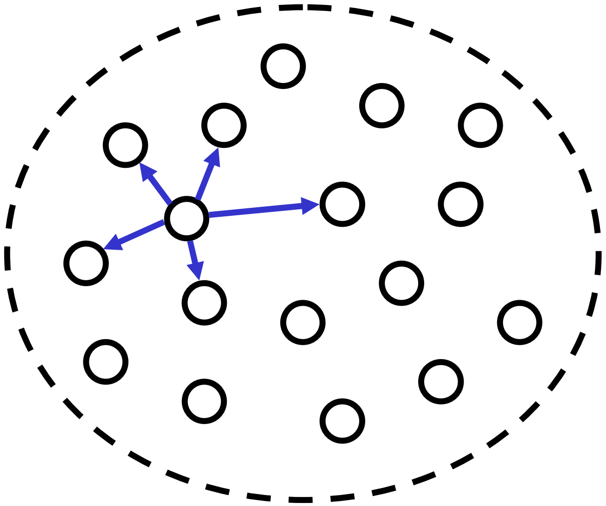


social networks

1. The Meta-Design of Complexity

➢ Geometric, regular networks (2-D, 3-D) **[TODAY'S FOCUS]**

Network	***Nodes***	***Edges***
 BZ reaction	*molecules*	*collisions*
 slime mold	*amoebae*	*cAMP*
 embryo	*cells*	*"morphogens"*
 insect colonies	*ants, termites*	*pheromone*
 flocking, traffic	*animals, cars*	*perception*
 swarm sync	*fireflies*	*photons* *± long-range*

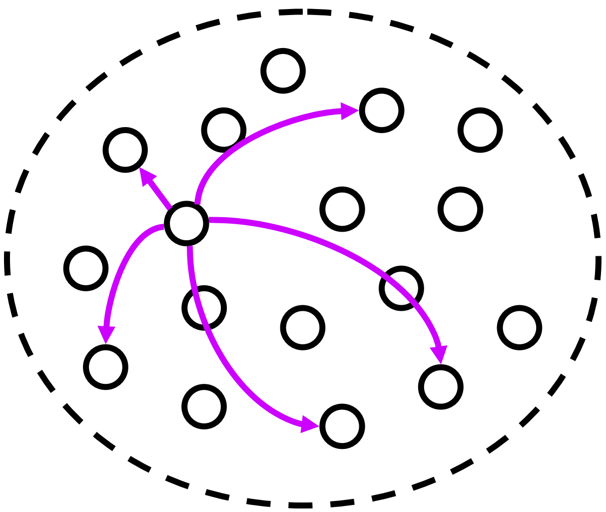


- interactions inside a local neighborhood in 2-D or 3-D geometric space
- limited "visibility" within Euclidean distance

1. The Meta-Design of Complexity

➢ Semi-geometric (spatial), irregular (non-spatial) networks

Network	***Nodes***	***Edges***
Internet	*routers*	*wires*
the brain	*neurons*	*synapses*
WWW	*pages*	*hyperlinks*
Hollywood	*actors*	*movies*
gene regulation	*proteins*	*binding sites*
ecosystems	*species*	*competition*



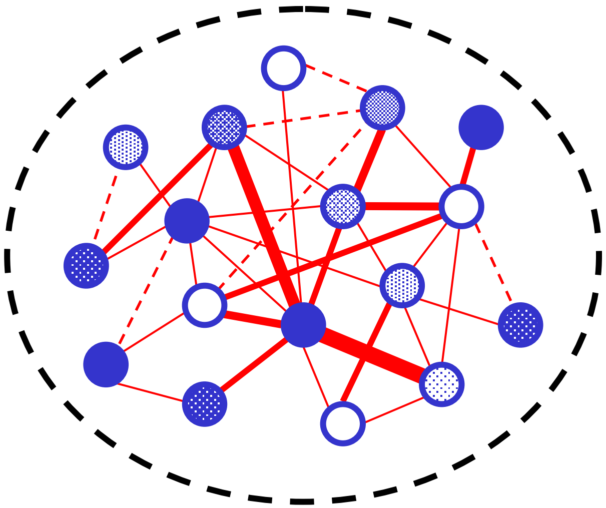
➢ still local neighborhoods, but with “long-range” links:

- either “element” nodes *located in space*
- or “categorical” nodes ***not located in space***

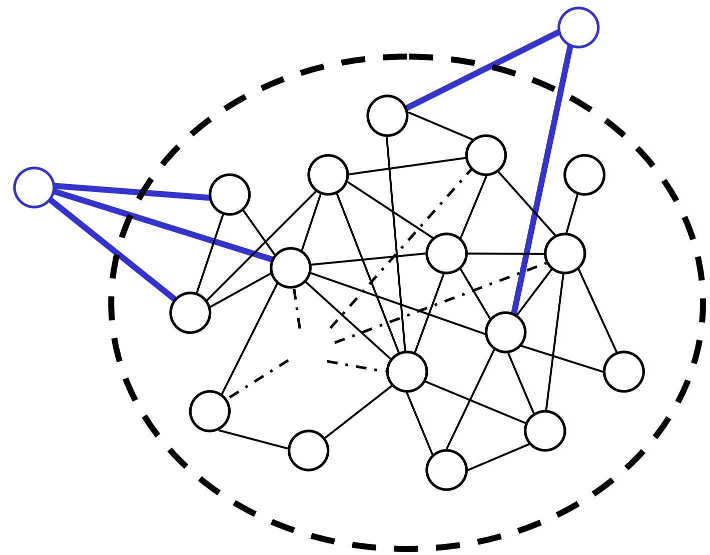
➢ still limited “visibility”, but not according to distance

1. The Meta-Design of Complexity

➢ Elementary features & network evolution



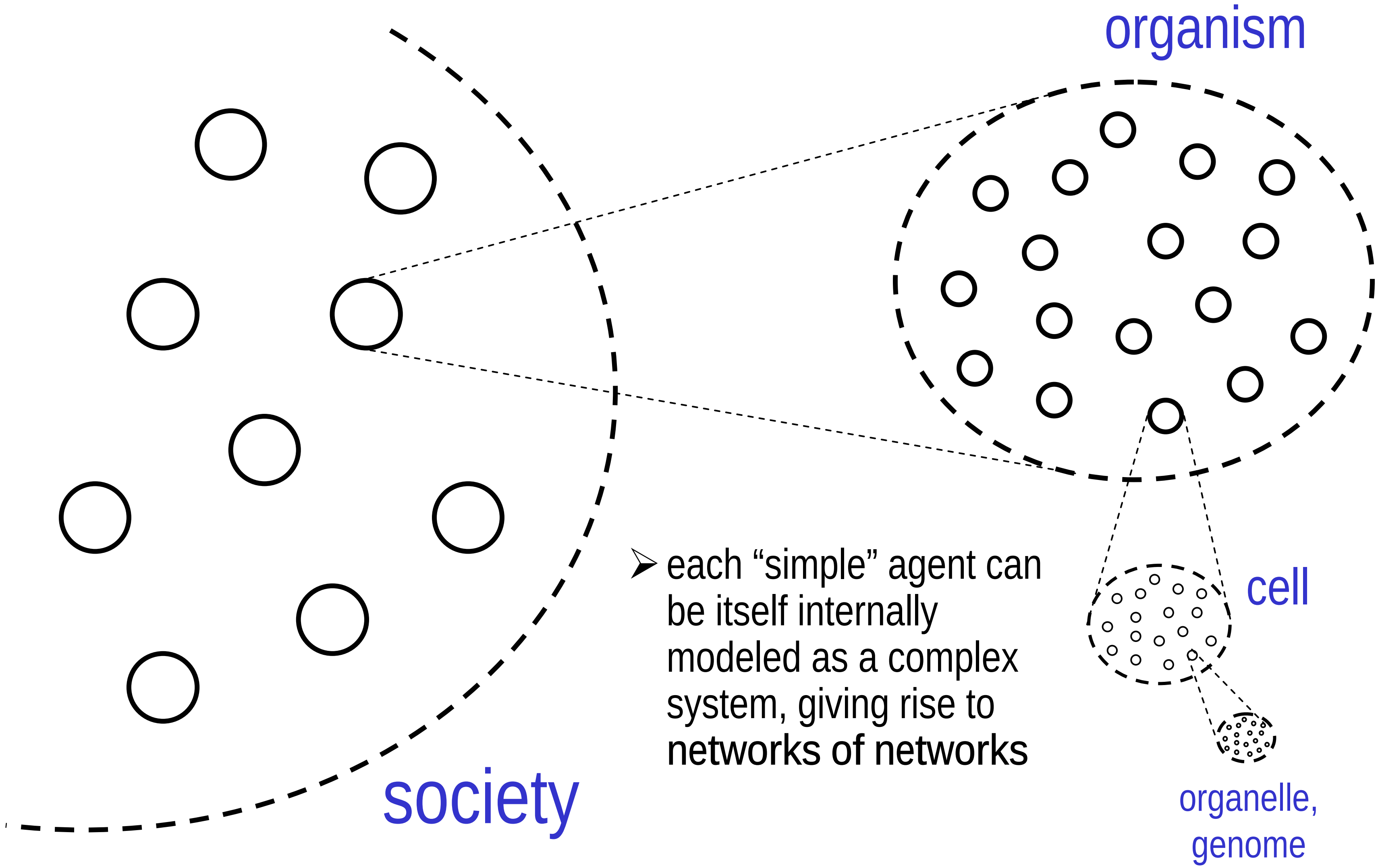
- the **state** of a network generally evolves on two time-scales:
  - fast time scale: node activities
  - slow time scale: connection weights
- examples:
  - neural networks: activities & learning
  - gene networks: expression & mutations



- the **structural complexity** of a network can also evolve by adding or removing nodes and edges
- examples:
  - Internet, WWW, actors. ecology, etc.

1. The Meta-Design of Complexity

- Multi-scale hierarchy of organizational levels



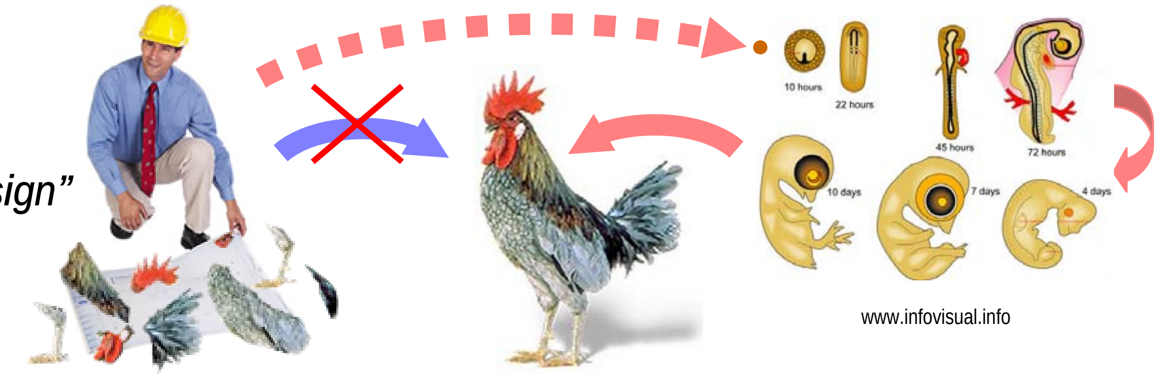
  - each “simple” agent can be itself internally modeled as a complex system, giving rise to **networks of networks**

1. The Meta-Design of Complexity

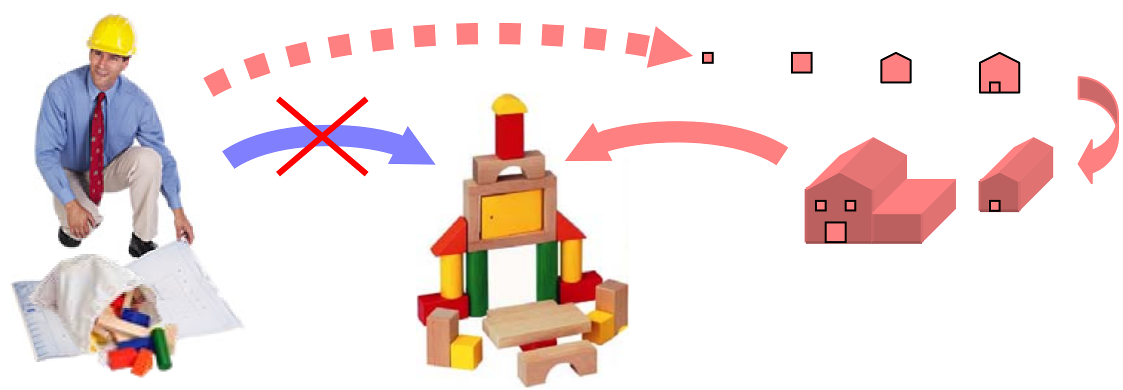
➢ From centralized heteronomy to decentralized autonomy

- ✓ artificial systems **are built** exogenously, organisms endogenously **grow**

replacing
systems design
with
systems *“meta-design”*

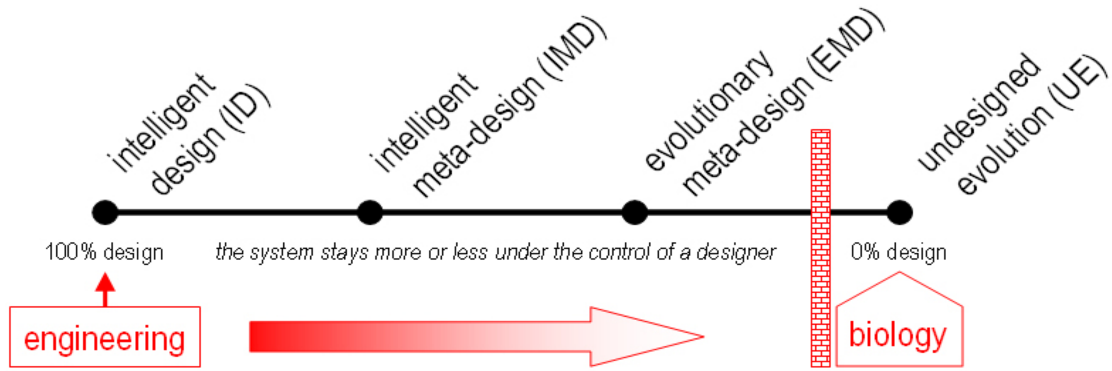


- ✓ future engineers should “step back” from their creation and only set the **generic** conditions for systems to self-assemble, self-regulate and evolve



1. The Meta-Design of Complexity

➢ Pushing engineering design toward evolutionary biology



intelligent "hands-on" design		intelligent & evolutionary "meta-design"
heteronomous order	▪	autonomous order
centralized control	▪	decentralized control
manual, extensional design	▪	automated, intentional design
engineer as a micromanager	▪	engineer as a lawmaker
rigidly placing components	▪	allowing fuzzy self-placement
tightly optimized systems	▪	hyperdistributed & redundant systems
sensitive to part failures	▪	insensitive to part failures
need to control	▪	prepare to adapt & self-regulate
need to redesign	▪	prepare to learn & evolve
complicated systems: planes, ~~computers~~	▪	*complex* systems: Web, market... computers?

1. The Meta-Design of Complexity

- Harnessing natural complex adaptive systems for ICT
  - ✓ natural complex adaptive systems, biological or social, can be a powerful new source of inspiration for future ICT in their shift toward autonomy
  - ✓ “emergent engineering” will be less about direct design and more about *developmental* and *evolutionary* meta-design
  - ✓ it will stress the importance of setting fundamental laws of development and *developmental variations*, before these variations can be selected
  - ✓ decentralized, unplanned “complex” systems could actually become the most economical and robust type of systems—in one word: the *simplest*
    - it is centralized, planned systems that are uniquely costly and fragile, as they require another intelligent system (us) to build and intervene into
  - ✓ many initiatives toward a convergence of “nano” *(swarm of components)*, “bio” *(complexity)*, “info” *(systems design)* and “cogno” *(intelligent systems)*
    - NBIC (US), FET & NEST (EU), “organic computing”, “amorphous computing”, “natural computation”, “pervasive computing”, “ambient intelligence”, etc.

Complex Systems Engineering from Biological Development

1. The Meta-Design of Complexity

2. Concrete Models Based on Multicellular Systems

    a. Embryomorphic Engineering

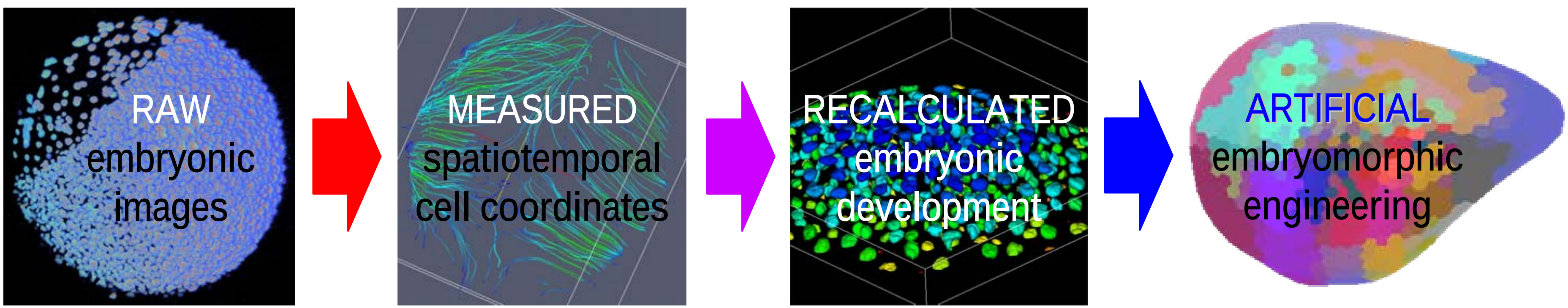
    b. Neurodynamic Pattern Recognition

    c. Immune Network Security

3. Planning for Autonomy

2.a Embryomorphic Engineering

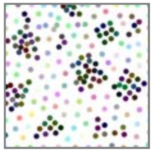
- Observing, modeling → exporting biological development
  - ✓ automating the **observation** and description of developing organisms with image processing, statistical and machine learning techniques
  - ✓ designing mathematical/computational **models** of embryonic growth
  - → ***implementing*** *biological development in engineering systems: distributed architectures as a prerequisite for evolutionary innovation*



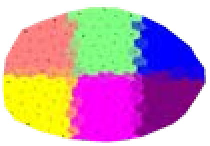
European projects “Embryomics” & “BioEmergences”

2.a Embryomorphic Engineering

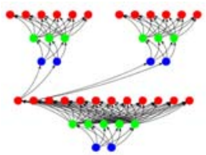
- The self-made puzzle: integrating *self-assembly* and *pattern formation* under non-random *genetic regulation*
  - ✓ self-assembly (SA)



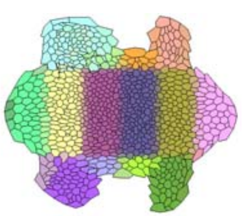
    - usually focuses on pre-existing components endowed with fixed shapes
    - . . . but cells *dynamically divide and differentiate* toward selective adhesion
  - ✓ pattern formation (PF)



    - generally orderly states of activity on top of continuous 2-D or 3-D substrate
    - . . . but gene expression patterning arises in *perpetually reshaping* organism
  - ✓ non-random genetic regulation (GRN)



    - both phenomena often thought stochastic: mixed components that randomly collide in SA; spots and stripes that pop up from instabilities in PF



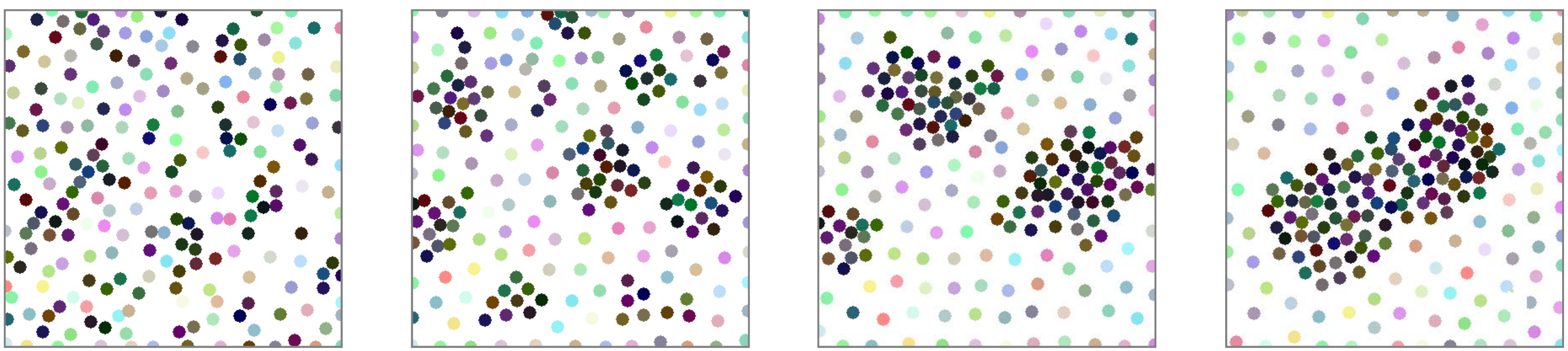
    - . . . but cells are *pre-positioned* where they divide, and genetic identity domains are *highly regulated* in number and position

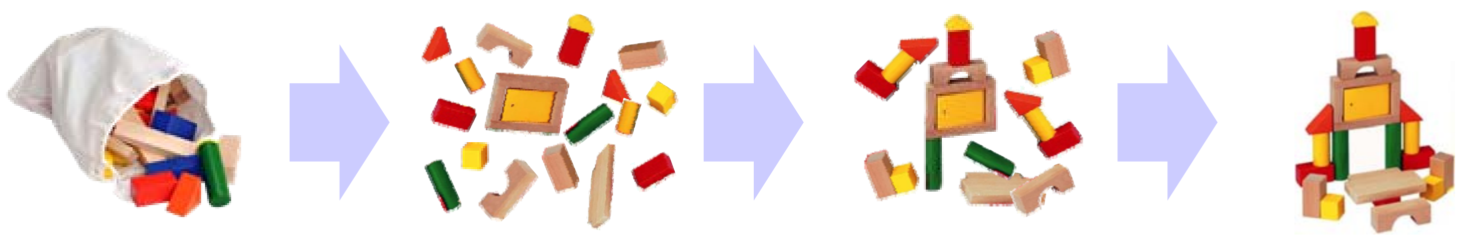
→ *integrate these 3 aspects in artificial “embryomorphic” systems*

Self-Assembly + Pattern Formation = Morphogenesis

- *Molecular*-style SA: structuration from a random mix
  - ✓ “shaking the puzzle box”
    - α particles randomly collide and cluster together within a sea of β particles
    - like molecules, dissociated cells can also spontaneously sort again
    - however, mostly artificial experiments; not a major natural mechanism

    $\rightarrow$ *the complex architecture of an organism does not emerge out of a giant swarm of trillions of disaggregated cells reassembling in parallel*

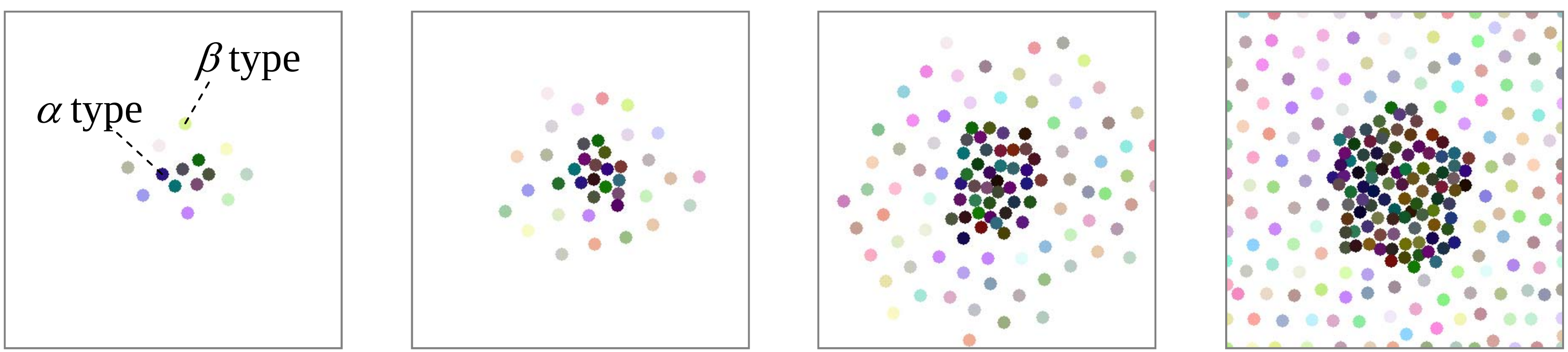


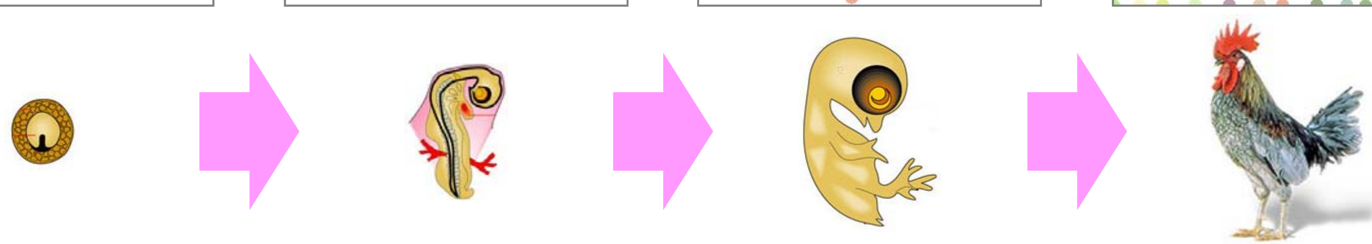


Self-Assembly + Pattern Formation = Morphogenesis

➢ *Multicellular*-style SA: structuration from development

- ✓ “growing the embryo”
  - starting with only a few particles of each type
  - particles *divide* into same-type particles, under uniform probability
  - new particles pop up *pre-positioned* near the type that produced them
  - particles only briefly rearrange within their local neighborhood





Self-Assembly + Pattern Formation = Morphogenesis

- Biological cells use mechanisms that greatly facilitate SA
  - ✓ future artificial systems design could follow a similar approach
    - instead of letting components haphazardly try to match each other's pre-existing constraints, like molecules in a solution. . .
    - . . . let components dynamically create and reshape themselves "on the spot," as cells do
  - ✓ from *stochastic* (molecular-style) self-assembly to *programmable* (multicellular-style) self-assembly
    - components must be able to dynamically modify their behavior (divide, differentiate, migrate) through *communication*
    - cells do not just snap into place; they send molecular signals to each other
  - → *cells form **patterns of differentiation** at the same time that they are self-assembling*

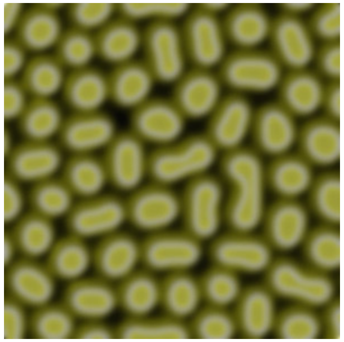
Self-Assembly + Pattern Formation = Morphogenesis

➢ Traditional PF is stochastic, biological PF is not

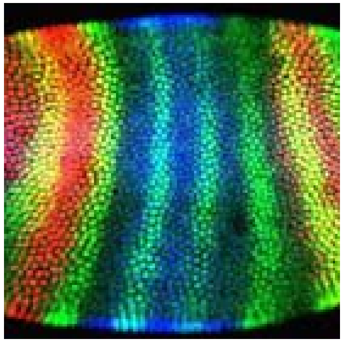
- randomness at micro-level (elts) and meso-level (patterns)
- PF research focuses on *instabilities* and amplification of fluctuations
- outcome generally *unpredictable* in number and position of domains
- conversely, macroscopic formation fairly regular: repeated motifs, statistical *uniformity* like textures

- mesoscopic organs and limbs have intricate, non-random morphologies
- reaction-diffusion based(?) animal coats are only a marginal aspect
- development is *reproducible* in number and position of body parts
- most of organism development is under deterministic genetic control: *heterogeneous*, rich in information

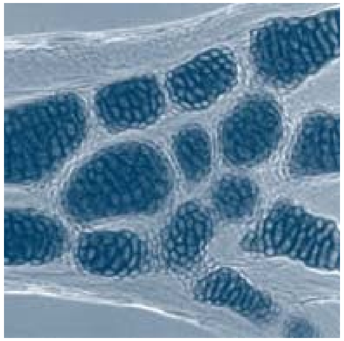
convection cells
www.chabotspace.org



reaction-diffusion
texturegarden.com/java/rd



fruit fly embryo
Sean Caroll, U of Wisconsin



larval axolotl limb
Gerd B. Müller

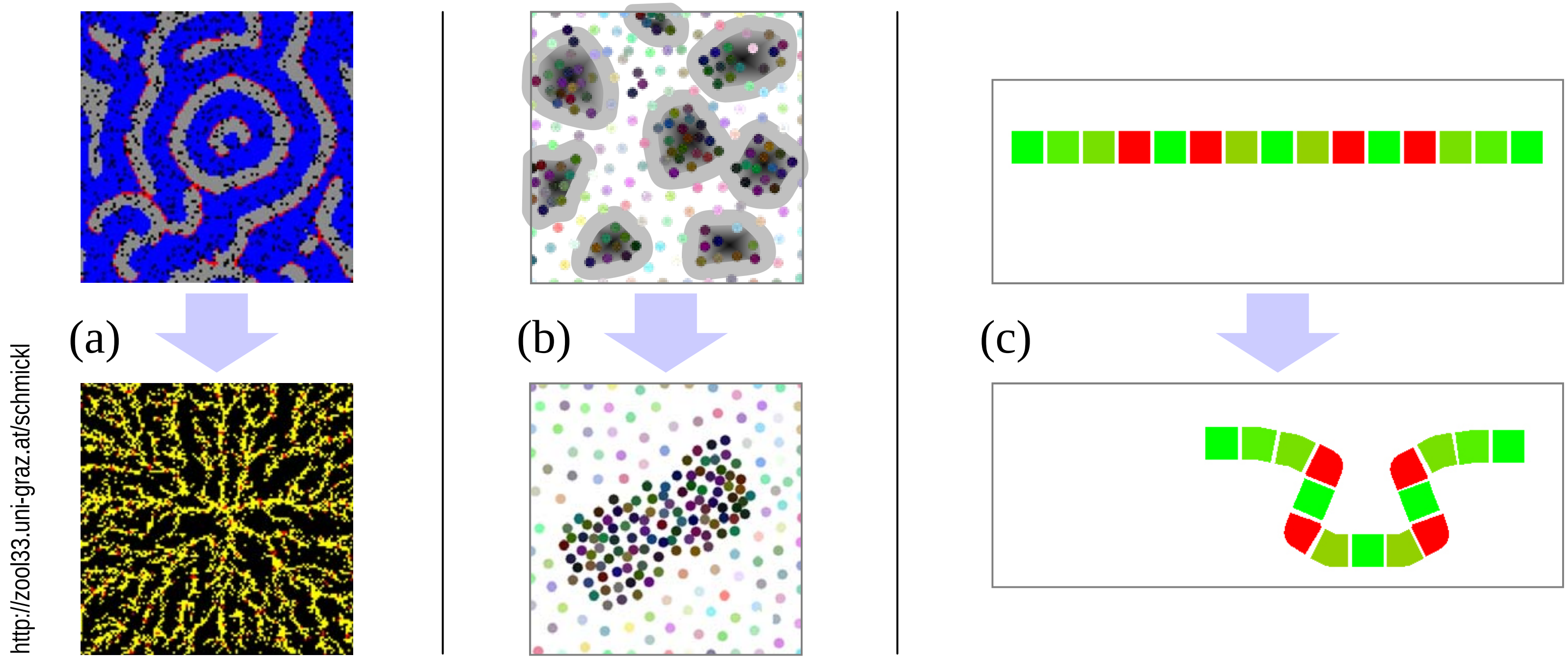
Self-Assembly + Pattern Formation = Morphogenesis

- Embryogenesis combines PF and morphogenetic SA
  - ✓ shapes from patterning; patterns from shaping
    - structures are “sculpted” from the self-assembly of elements, prompted by the “painting” of their genetic identity
    - conversely, newly formed shapes are able to support, and trigger, new domains of genetic expression
  - ✓ tightly integrated loop under non-random genetic regulation
    - DNA is “consulted” at every step of this exchange, in every cell
    - it produces the proteins that guide the cell’s highly specific biomechanic behavior (shaping) and signalling behavior (patterning)

Self-Assembly + Pattern Formation = Morphogenesis

➢ “Shape from patterning” examples

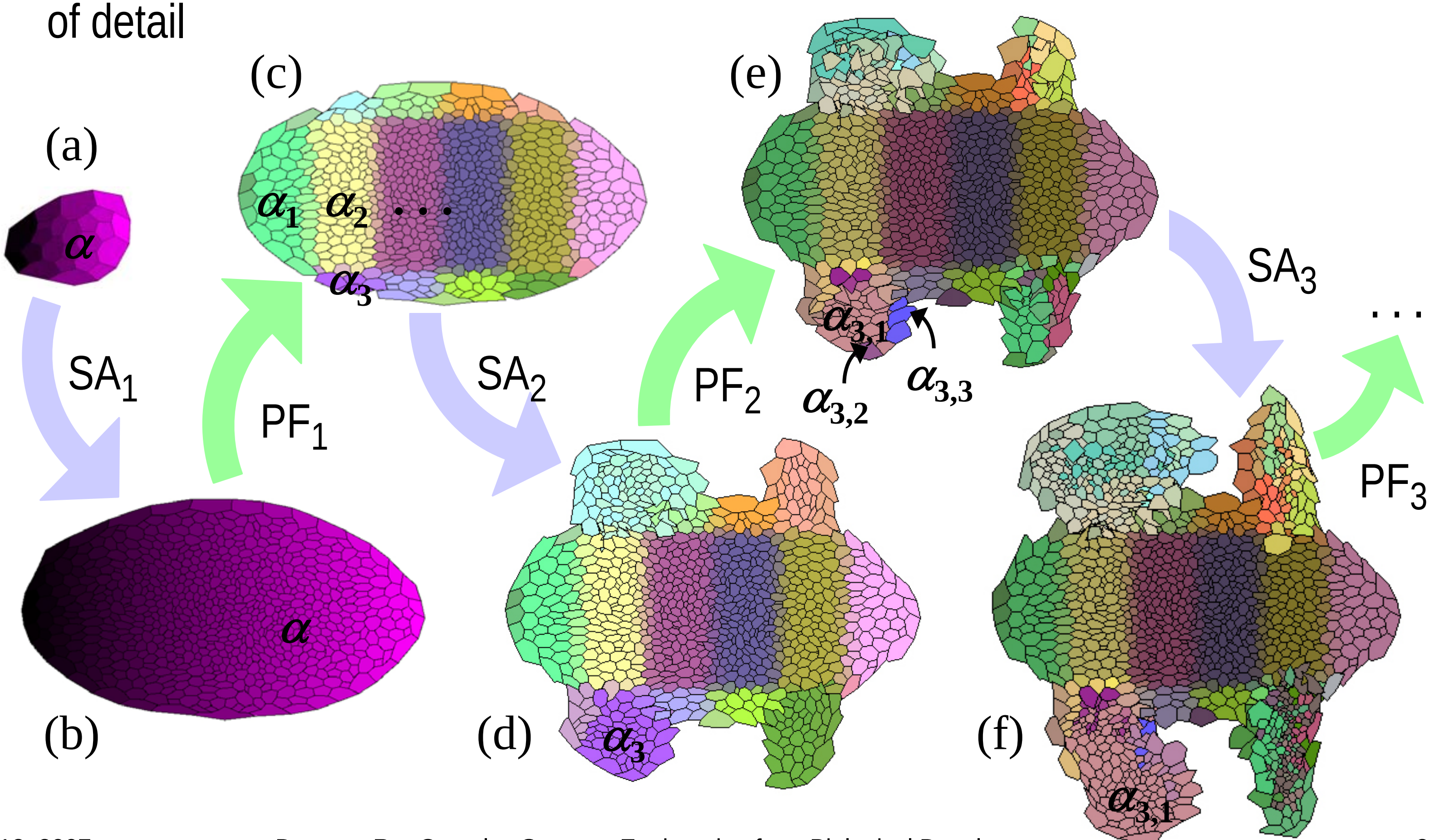
- ✓ deriving morphogenetic SA (bottom frames) from PF (top frames)
  - a) slime mold amoebae first generate waves of chemical signalling (top), then follow concentration gradients and aggregate (bottom)
  - b) type-α particles differentiating from a prepattern before assembling
  - c) bending angle of each γ particle also determined by a prepattern of identity



Self-Assembly + Pattern Formation = Morphogenesis

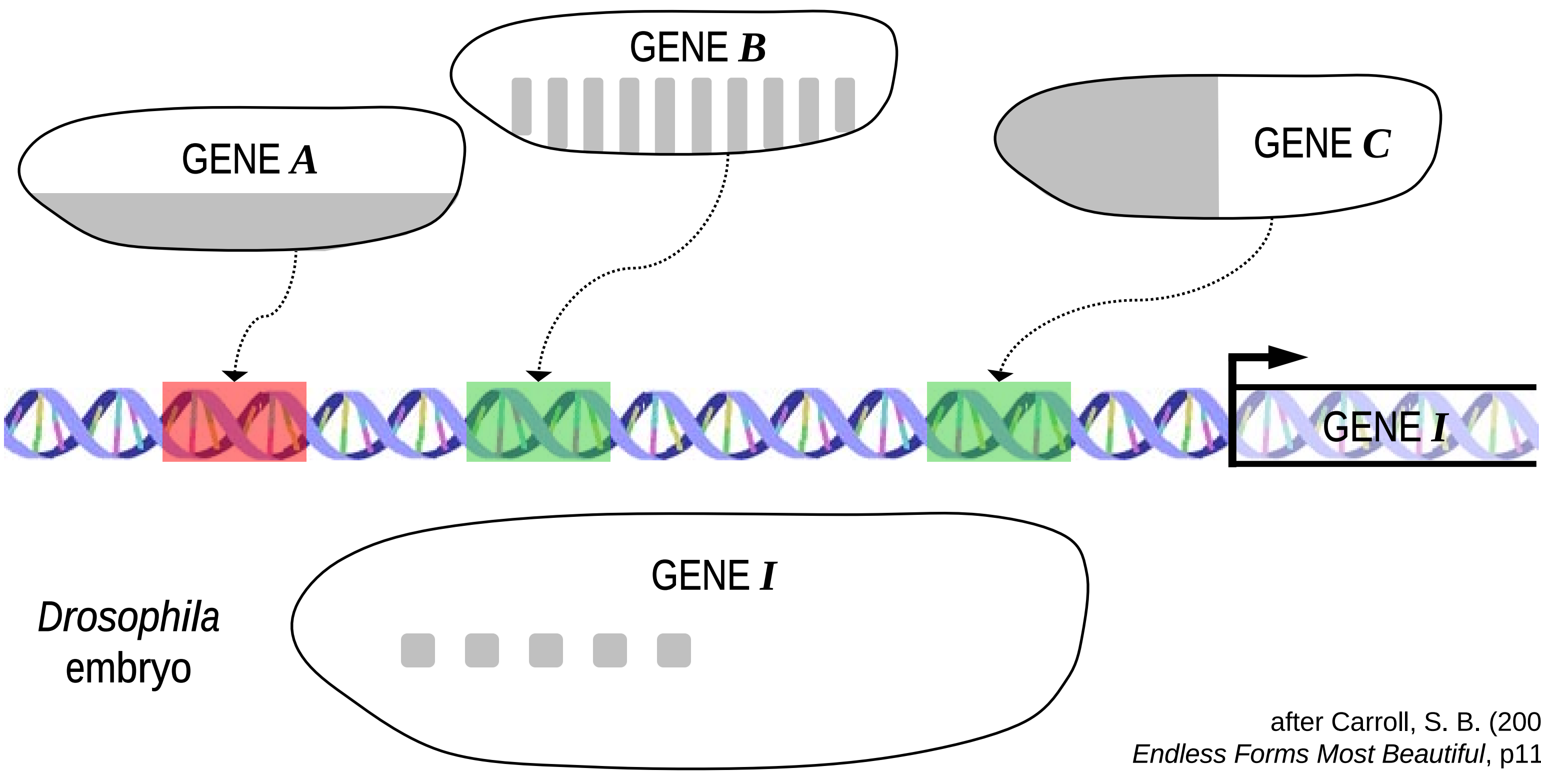
➢ Embryomorphic architectures

- *functional dependency between cell identities and mechanical cell behaviors*
- alternation of PF-induced differentiation and heterogeneous-type SA at all scales of detail



Self-Assembly + Pattern Formation = Morphogenesis

- Developmental genes are expressed in spatial domains
  - ✓ thus combinations of switches can create patterns by union and intersection, for example: $I = (\text{not } A) \text{ and } B \text{ and } C$

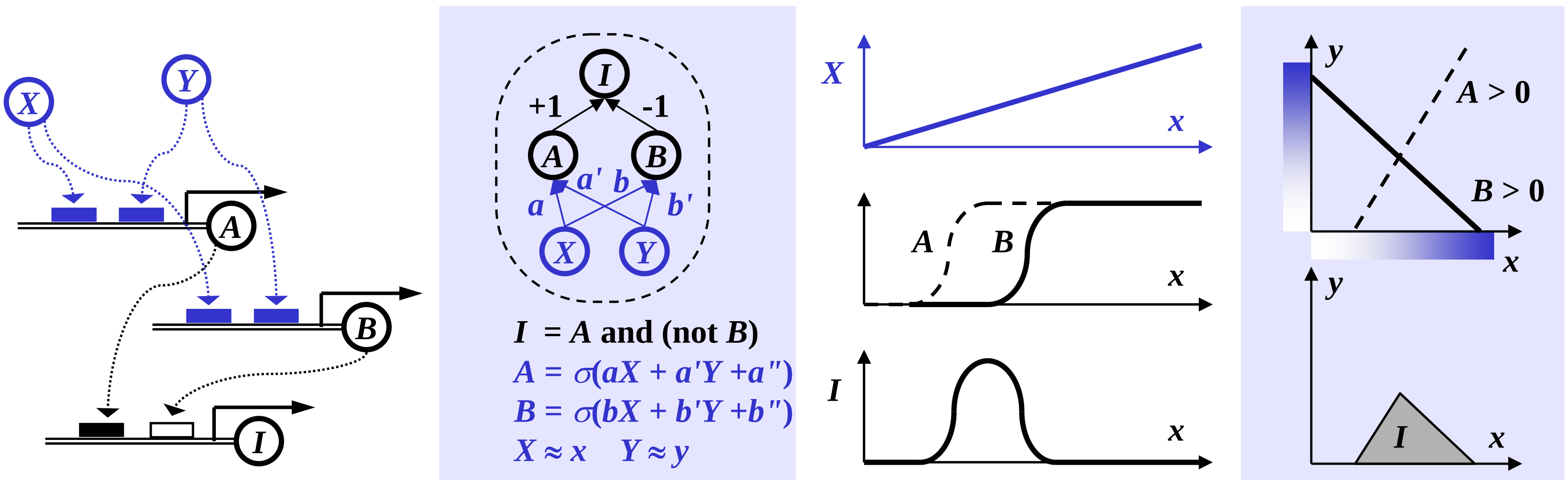


after Carroll, S. B. (2005) *Endless Forms Most Beautiful*, p117

Self-Assembly + Pattern Formation = Morphogenesis

➢ Three-tier GRN model: integrating positional gradients

✓ A and B are themselves triggered by proteins X and Y

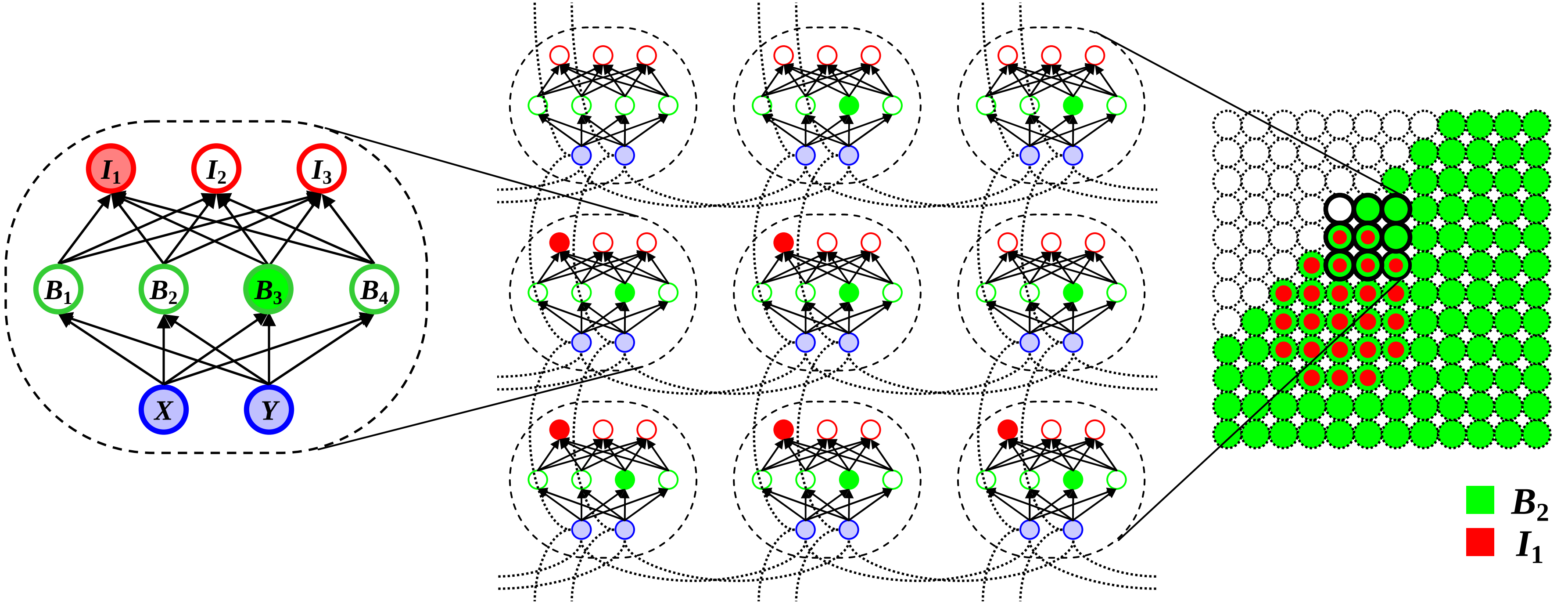


✓ X and Y diffuse along two axes and form concentration gradients

→ *different thresholds of lock-key sensitivity create different territories of gene expression in the geography of the embryo*

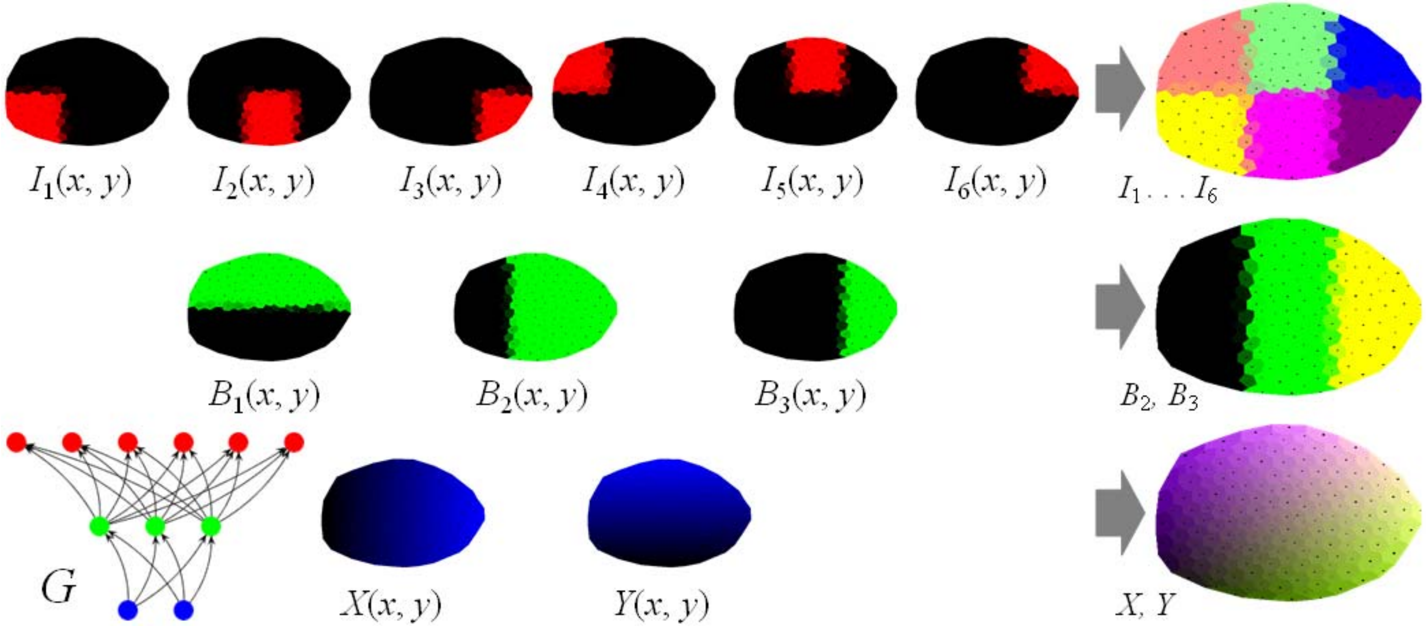
Self-Assembly + Pattern Formation = Morphogenesis

- A lattice of Positional-Boundary-Identity (PBI) GRNs
  - ✓ network of networks: each GRN is contained in a cell, coupled to neighboring cells via the positional nodes (for diffusion)
  - ✓ a pattern of gene expression is created on the lattice



Self-Assembly + Pattern Formation = Morphogenesis

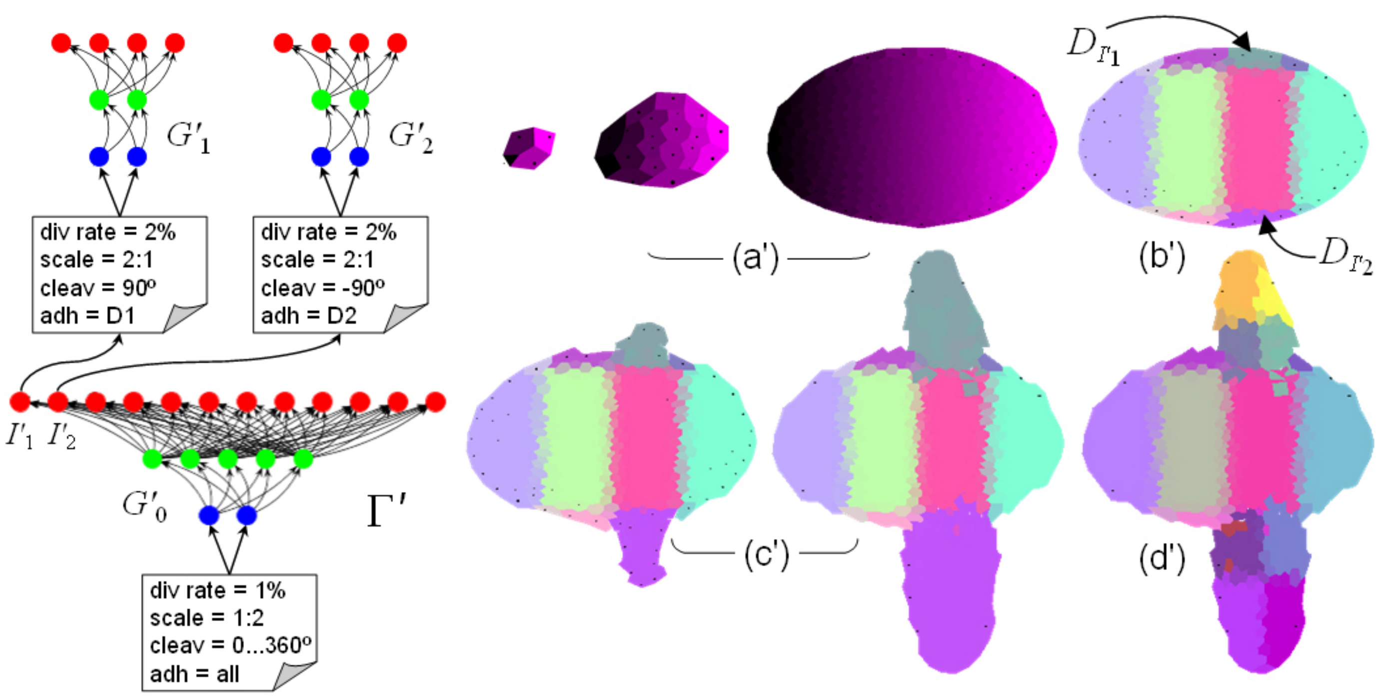
- **The hidden geography of the embryo**
  - ✓ self-patterning obtained from a 3B-6I gene regulatory network *G* in a 200-cell oval-shaped embryo
  - ✓ each view is “dyed” for the expression map of one of the 11 genes, e.g.: $B_1 = \sigma(Y - 1/2)$, $B_2 = \sigma(X - 1/3)$, $I_6 = B_1 B_3$ ...



Self-Assembly + Pattern Formation = Morphogenesis

➢ Inhomogeneous cell division and adhesion

- ✓ using differential adhesion, anisotropic cleavage planes and rescaling, this model can generate directional offshoot akin to limb development



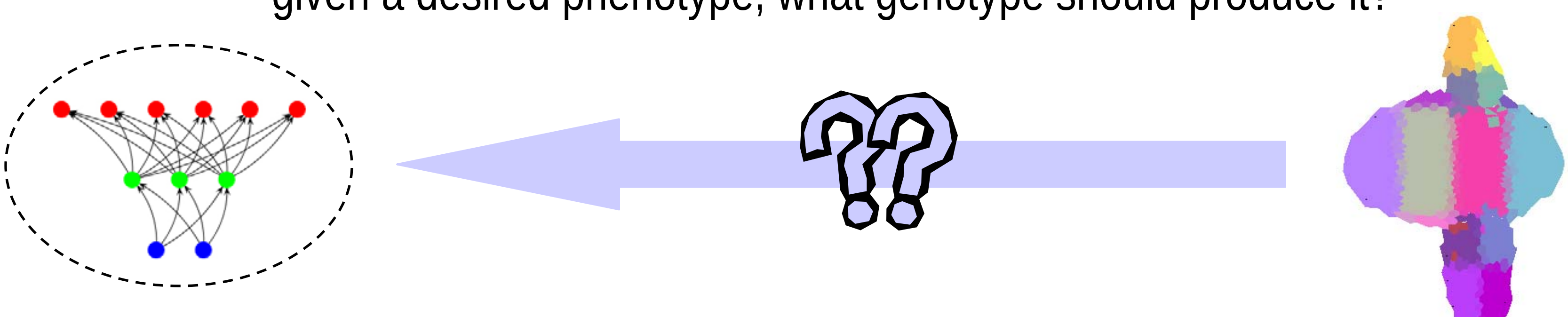
- ✓ here, different weights in base module G'_0 make a thicker central row, and place I'_1 and I'_2 dorsally and ventrally
- ✓ different adhesion coefficients also make I'_1 and I'_2 grow “limbs”, sub-patterned by G'_1 and G'_2

Complex Systems Engineering from Biological Development

1. The Meta-Design of Complexity
2. Concrete Models Based on Multicellular Systems
    a. Embryomorphic Engineering
    b. Neurodynamic Pattern Recognition
    c. Immune Network Security
3. Planning for Autonomy

3. Planning for Autonomy

- **The paradoxical goals of complex systems engineering**
  - how can we expect specific characteristics from systems that are otherwise free to invent themselves?
    - how to plan self-organization?
    - how to control decentralization?
    - how to design evolution?
  - the challenge is not so much to *allow* self-organization and emergence but, more importantly, to **guide** them
  - ex: embryomorphic engineering:
    - given a desired phenotype, what genotype should produce it?



3. Planning for Autonomy

- 3 challenges of CS engineers: growth, function, evolution

1. how does the system **grow**? (task of the developmental IMD engineer)
   - development results from a combination of elementary mechanisms: elements change internal state, communicate, travel, divide, die, etc.
   - starting from a single element, a complex and organized architecture develops by repeatedly applying these rules inside each element

   → *task 1 consists of combining these principles and designing their dynamics*

2. how does the system **function**? (task of the functional IMD engineer)
   - this task is about defining the nature of the elements their functionality: nano/bio components? software modules? robot parts? swarm robots?
   - are they computing? physically moving? or both? etc.

parameters = "genetic code"

3. how does the system evolve? (task of the EMD engineer)...
   - how the system varies (randomly)
   - how it is selected (nonrandomly)

3. Planning for Autonomy

➢ Selecting without expectations?

✓ different degrees of fitness constraints

a) selecting for a specific **organism** (shape, pattern)

- reverse problem: given the phenotype, what should be the genotype?
- **direct** recipe; ex: Nagpal's macro-to-microprogram Origami compilation
- otherwise: **learn** or **evolve** under strict fitness $\rightarrow$ difficult to achieve!

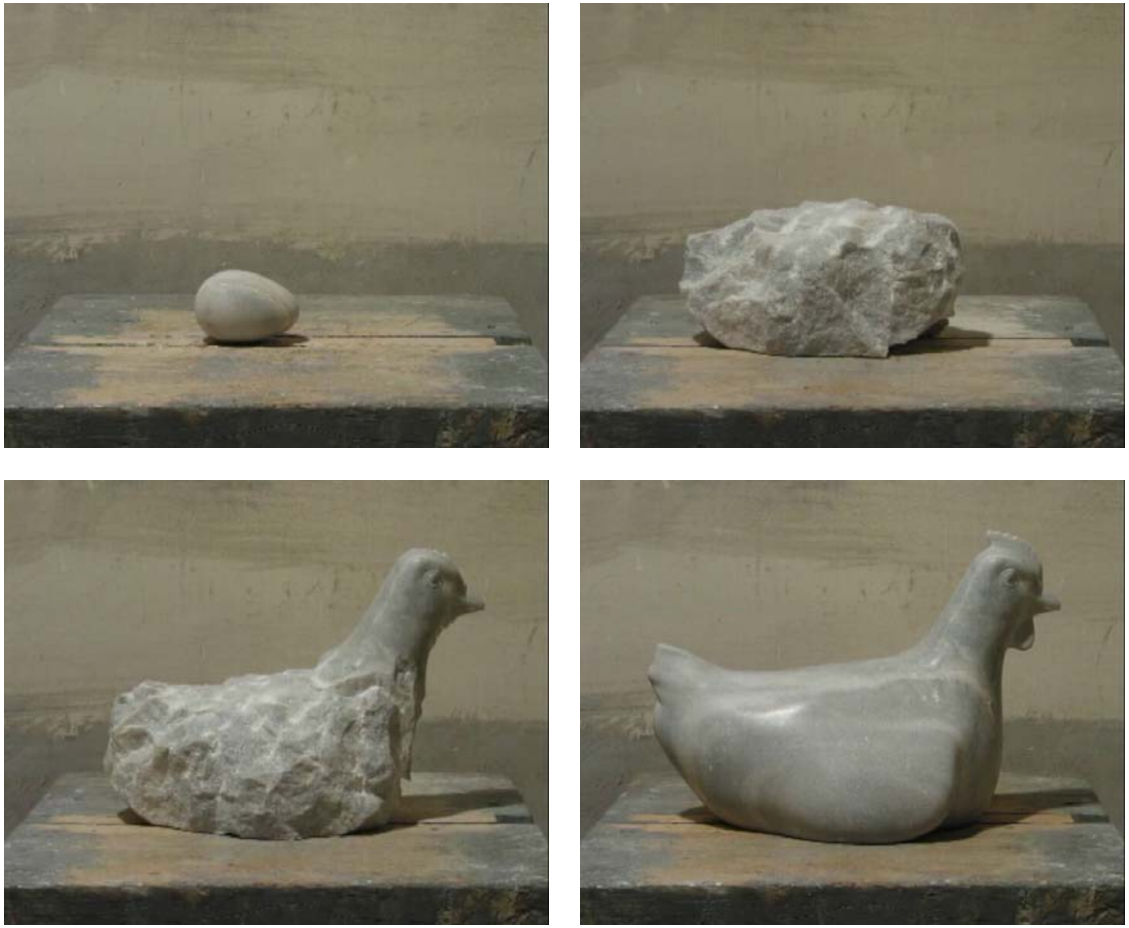
b) selecting for a specific **function**, leaving freedom of architecture

- given a task, optimize performance (computing, locomotion, etc.)
- be surprised by pattern creativity; ex: Avida, GOLEM, Framsticks

c) selecting the **unexpected**

- create a "solution-rich" space by (a) combinatorial tinkering on redundant parts and (b) relaxing/diversifying the requirements
- harvest interesting or surprising organisms from a free-range menagerie

Complex Systems Engineering from Biological Development



Ádám Szabó, *The chicken or the egg* (2005)
http://www.szaboadam.hu