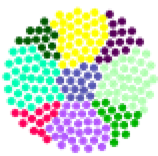
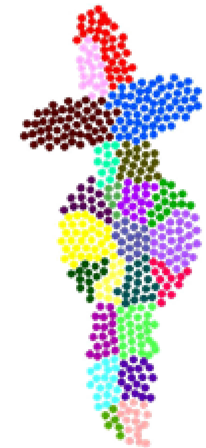
The self-organization and variability of complex modular architectures as a prerequisite to evolutionary innovation





René Doursat

http://doursat.free.fr

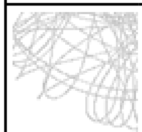
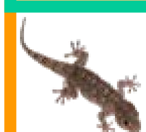
From flocks to shapes

Toward Morphogenetic Engineering

1. Self-organized *and* structured complex systems
2. Toward "evo-devo" engineering
3. A model of programmable morphogenesis
4. Evolutionary meta-design
5. Extension: programmable complex networks

From “statistical” to “morphological” complex systems

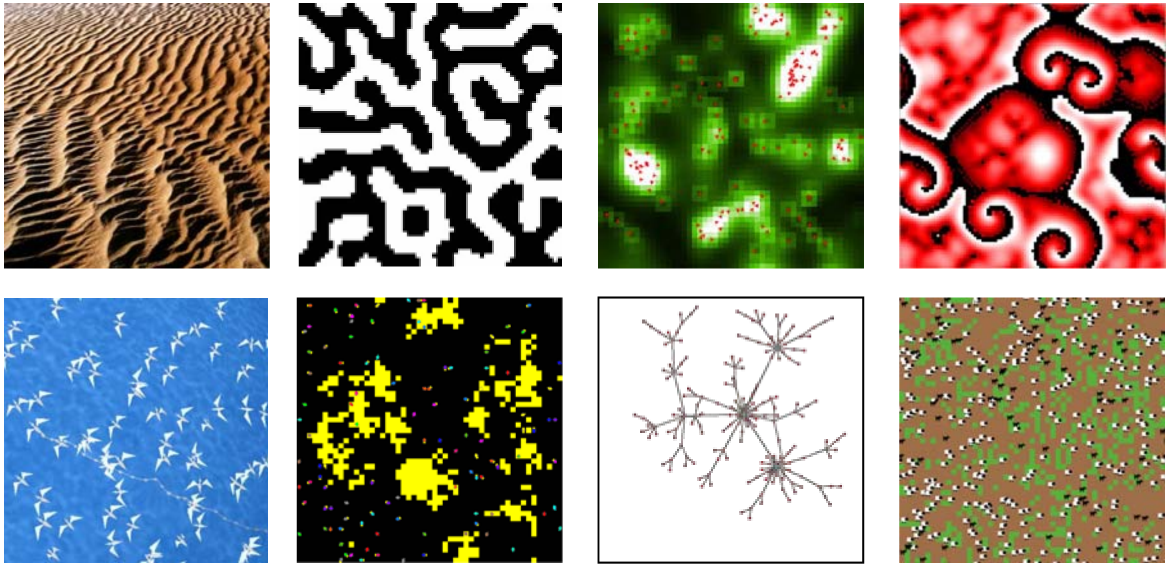
➢ A brief taxonomy of systems

Category	*Agents / Parts*	*Local Rules*	*Emergent Behavior*	*A “Complex System”?*
two-body problem	*few*	*simple*	*simple*	*NO*
 three-body pb, low-D chaos	*few*	*simple*	*complex*	*NO – too small*
 crystal, gas	*many*	*simple*	*simple*	*NO – few params suffice to describe it*
 patterns, swarms, complex networks	*many*	*simple*	*“complex”*	*YES – but mostly random and uniform*
 structured morphogenesis	*many*	*sophisticated*	*complex*	*YES – reproducible and heterogeneous*
 crowds with leaders, machines	*many*	*sophisticated*	*“simple”*	*COMPLICATED – not self-organized*

Statistical (self-similar) systems

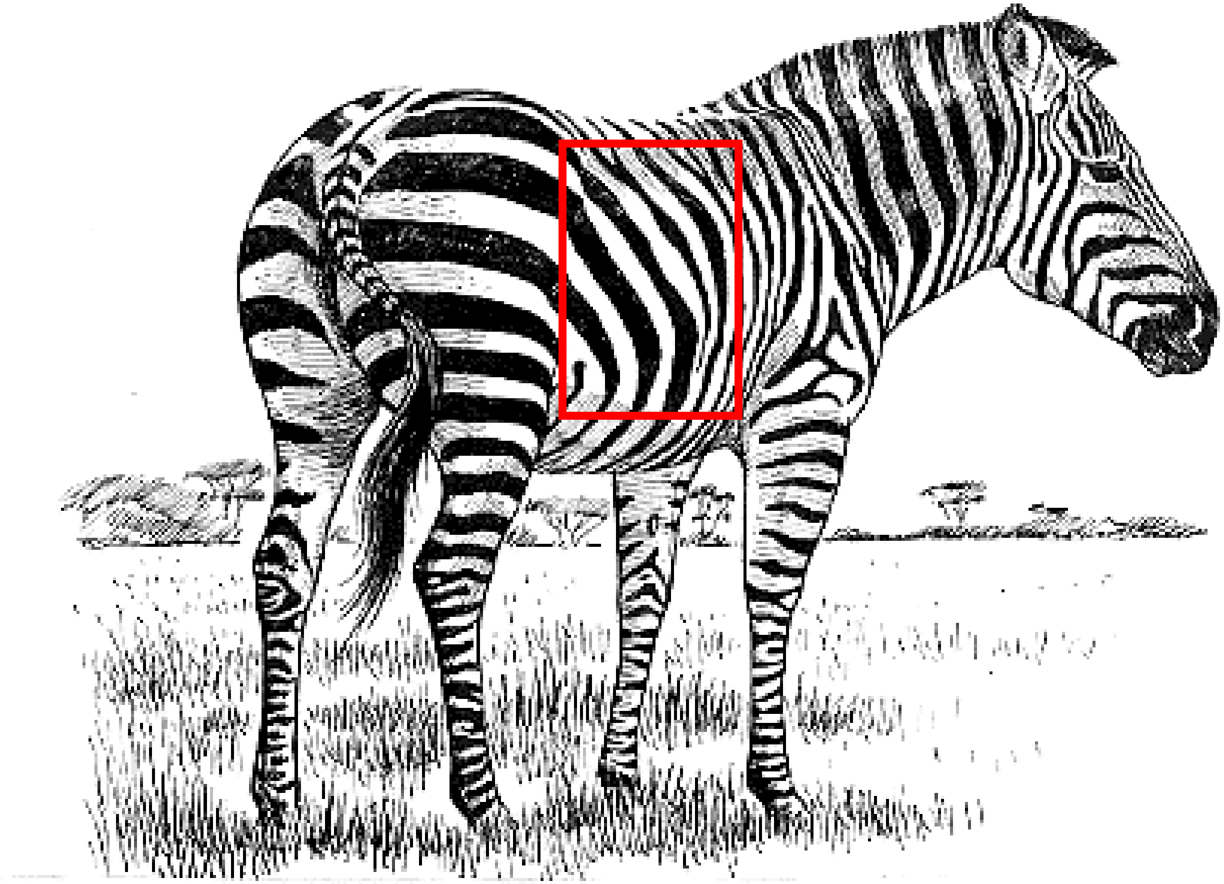
- Many agents, simple rules, "complex" emergent behavior

  → *the "clichés" of complex systems: diversity of pattern formation (spots, stripes), swarms (clusters, flocks), complex networks, etc.*



- ✓ yet, often like "textures": repetitive, statistically ***uniform***, information-poor
- ✓ spontaneous order arising from amplification of ***random*** fluctuations
- ✓ ***unpredictable*** number and position of mesoscopic entities (spots, groups)

Morphological (self-dissimilar) systems



“I have the stripes, but where is the zebra?”

—(attributed to) A. Turing, after his 1952 paper on morphogenesis

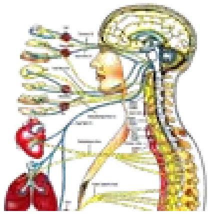
Morphological (self-dissimilar) systems

- Many agents, sophisticated rules, complex emergence
  - → *natural ex: organisms (cells)*

plants

vertebrates

arthropods



humans

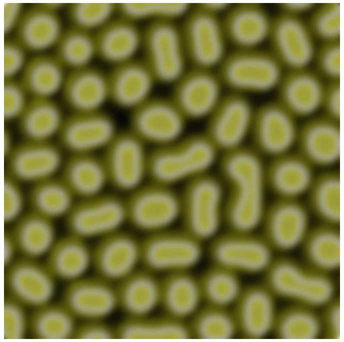
- ✓ mesoscopic organs and limbs have intricate, ***nonrandom*** morphologies
- ✓ development is highly ***reproducible*** in number and position of body parts
- ✓ ***heterogeneous*** elements arise under information-rich genetic control

- Biological organisms are self-organized *and* structured
  - ✓ because agent rules are more “sophisticated”: they can depend on the agent’s ***type*** and/or ***position*** in the system
  - ✓ the outcome (development) is truly complex but, paradoxically, can also be more ***controllable*** and ***programmable***

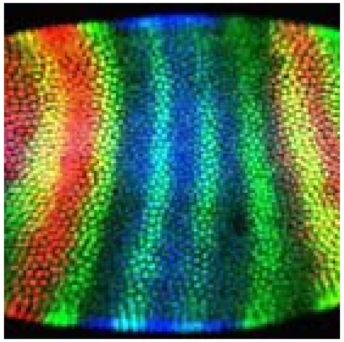
Statistical vs. morphological systems

- Physical pattern formation is *free*, biological PF is *guided*

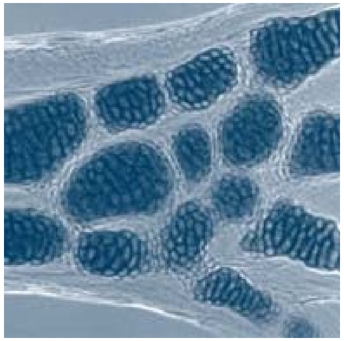
convection cells
www.chabotspace.org



reaction-diffusion
texturegarden.com/java/rd



fruit fly embryo
Sean Caroll, U of Wisconsin

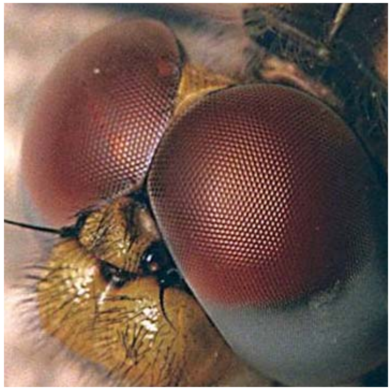


larval axolotl limb
Gerd B. Müller

Statistical vs. morphological systems

- ➢ Biotic forms combine a bit of "free" with a lot of "guided"
  - ✓ domains of free pattern embedded in a guided morphology

spots, stripes in skin
angelfish, www.sheddaquarium.org



ommatidia in eye
dragonfly, www.phy.duke.edu/~hsg/54

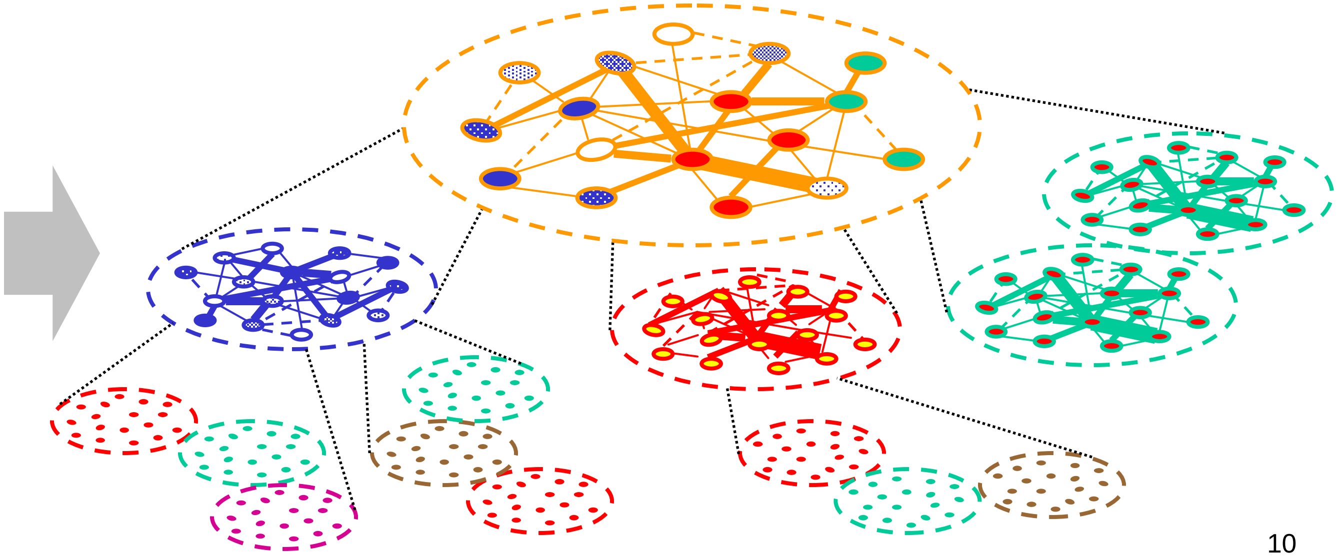
  - ✓ repeated copies of a guided form, distributed in free patterns

flowers in tree
cherry tree, www.phy.duke.edu/~fortney

segments in insect
centipede, images.encarta.msn.com

Beyond statistics: heterogeneity, modularity, reproducibility

- Complex systems can be much more than a "soup"
  - ✓ "complex" doesn't necessarily imply "homogeneous"...
    → ***heterogeneous** agents and diverse patterns, via positions*
  - ✓ "complex" doesn't necessarily imply "flat" (or "scale-free")...
    → ***modular**, hierarchical, detailed architecture (at specific scales)*
  - ✓ "complex" doesn't necessarily imply "random"...
    → ***reproducible** patterns relying on programmable agents*



Toward Morphogenetic Engineering

1. Self-organized *and* structured complex systems

2. Toward "evo-devo" engineering

3. A model of programmable morphogenesis

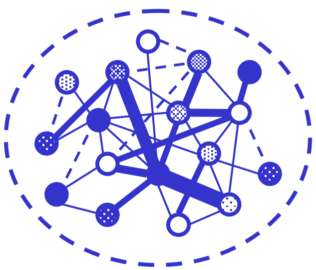
4. Evolutionary meta-design

5. Extension: programmable complex networks

Complex systems research

- The challenges of complex systems (CS) research

Transfers
- among systems

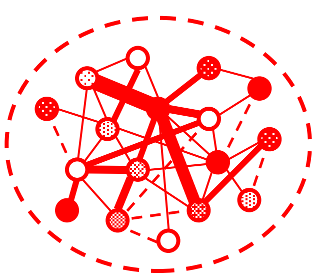


CS science: understanding “natural” CS

(i.e. spontaneously emergent, including human activity)

Exports
- decentralization
- autonomy, homeostasis
- learning, evolution

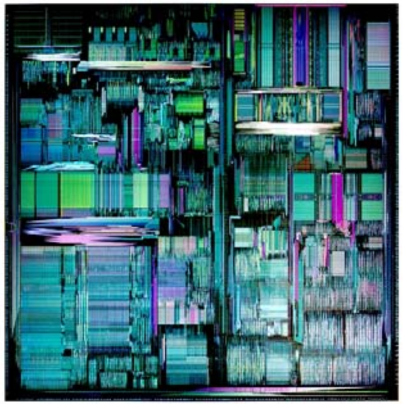
Imports
- observe, model
- control, harness
- design, use

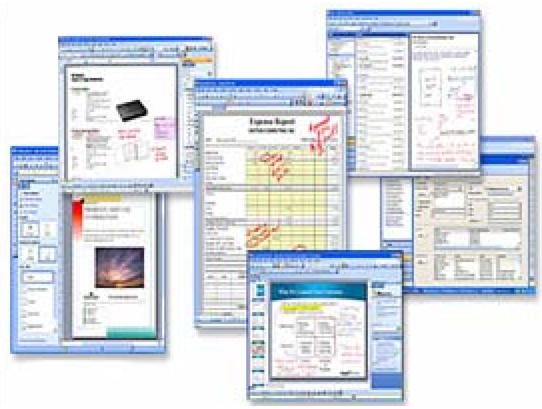


CS engineering: designing a new generation of “artificial” CS (i.e. harnessed, including nature)

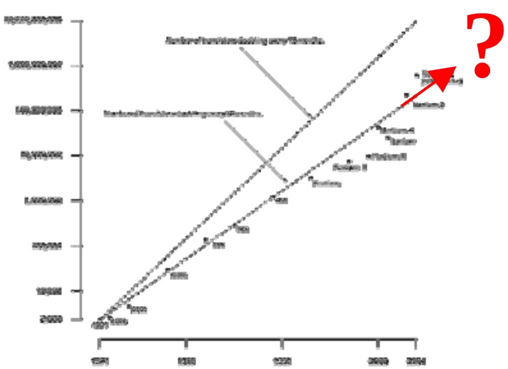
Complexity in ICT systems

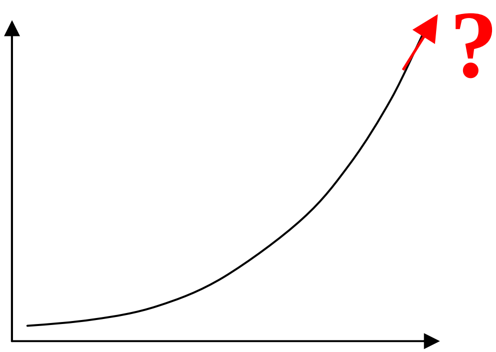
➢ Ineluctable breakup into myriads of modules/components,

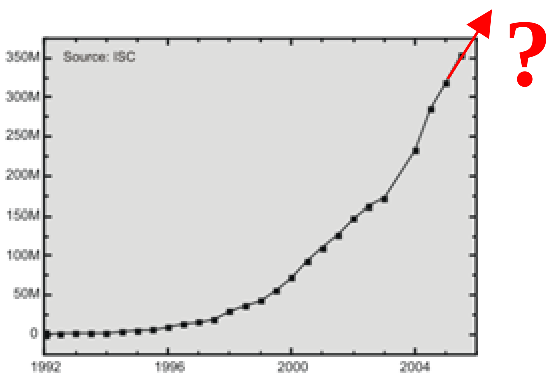




in hardware, software, or networks, ...







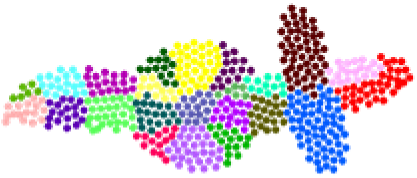
number of transistors/year

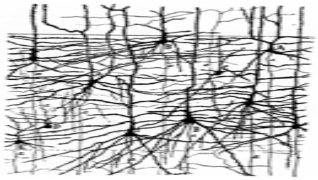
number of O/S lines of code/year

number of network hosts/year

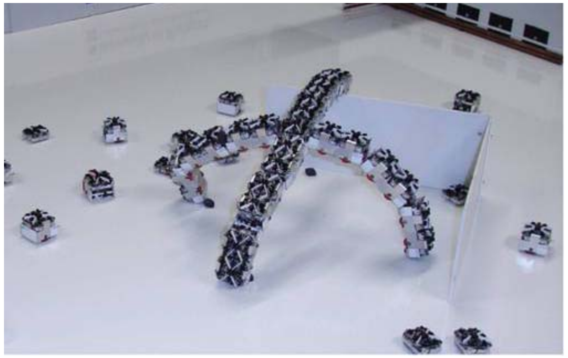
The need for morphogenetic abilities: self-architecturing

- Model natural systems → transfer to artificial systems
  - ✓ need for morphogenetic abilities in biological modeling
    - organism development
    - brain development



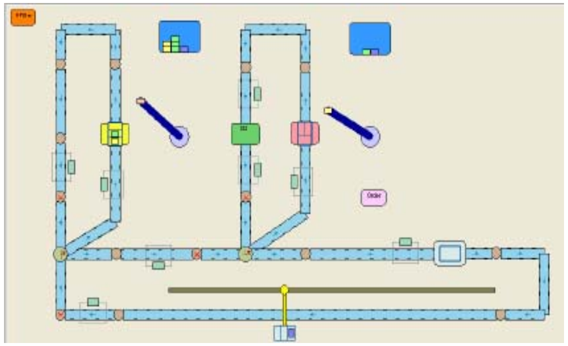


  - ✓ need for morphogenetic abilities in computer science & AI
    - self-forming robot swarm
    - self-architecturing software
    - self-connecting micro-components



http://www.symbrion.eu

  - ✓ need for morphogenetic abilities in techno-social eNetworked systems
    - self-reconfiguring manufacturing plant
    - self-stabilizing energy grid
    - self-deploying emergency taskforce

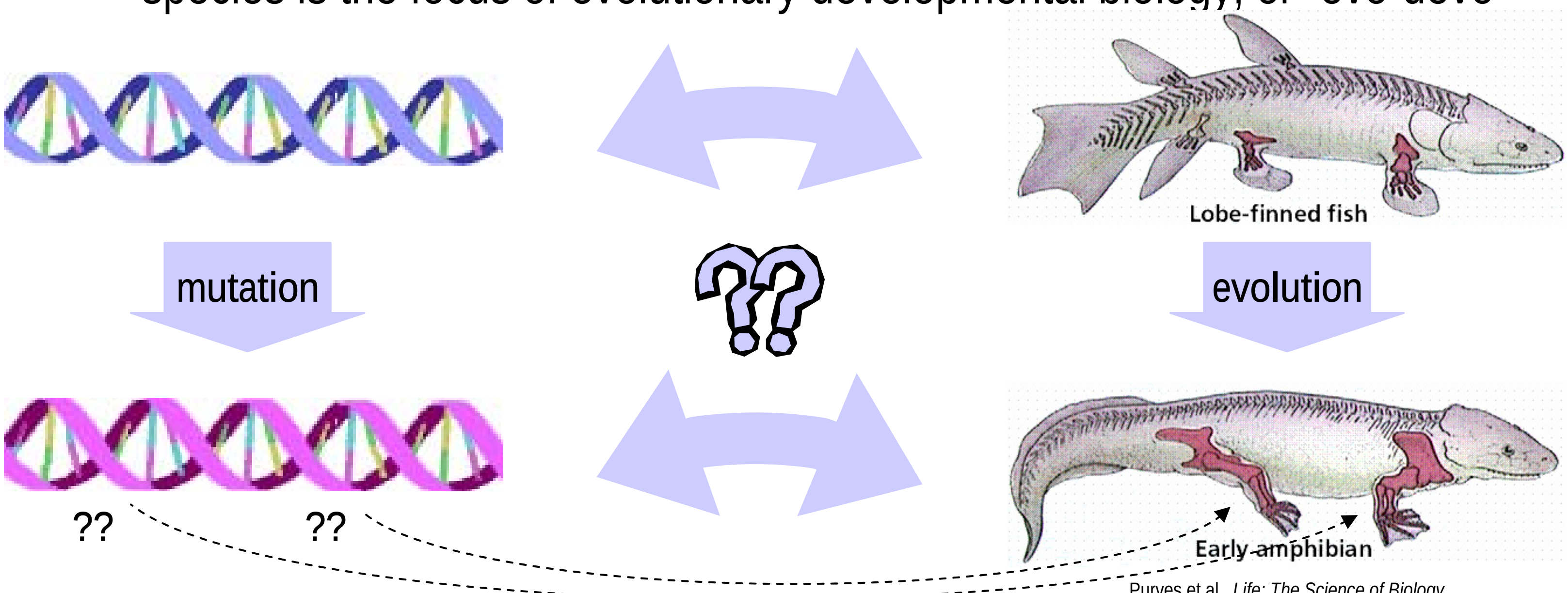


MAST agents, Rockwell Automation Research Center
{pvrba, vmarik}@ra.rockwell.com

2. Toward "evo-devo" engineering

➢ Development: the missing link of the Modern Synthesis

- ✓ biology's "Modern Synthesis" demonstrated the existence of a fundamental correlation between genotype and phenotype, yet the molecular and cellular mechanisms of development are still unclear
- ✓ the genotype-phenotype link cannot remain an abstraction if we want to unravel the generative laws of development and evolution
- ✓ understanding variation by comparing the actual development of different species is the focus of evolutionary developmental biology, or "evo-devo"



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

2. Toward "evo-devo" engineering

"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**

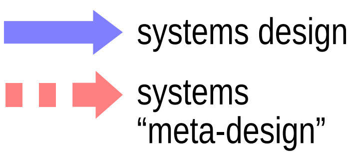
2. Toward “evo-devo” engineering

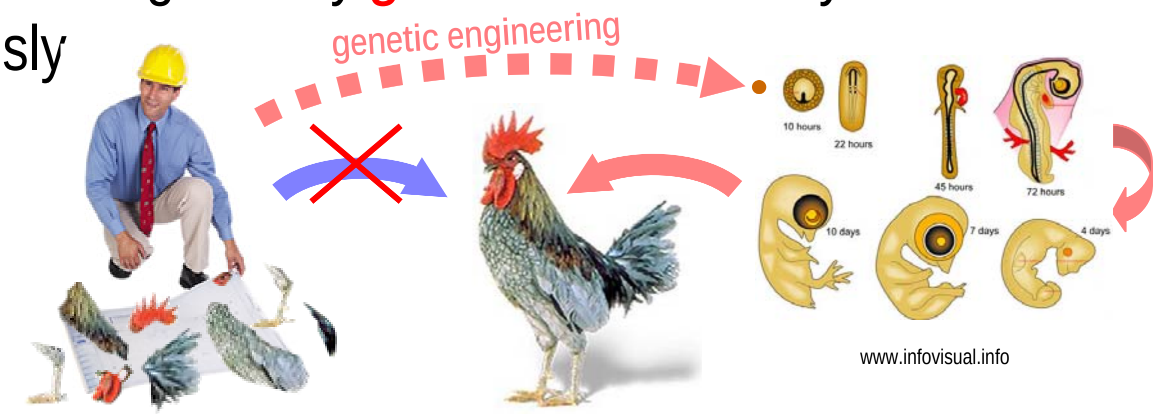
<u>How</u> does a static, nonspatial genome dynamically unfold in time and 3-D space?

<u>How</u> are morphological changes correlated with genetic changes?

2. Toward “evo-devo” engineering

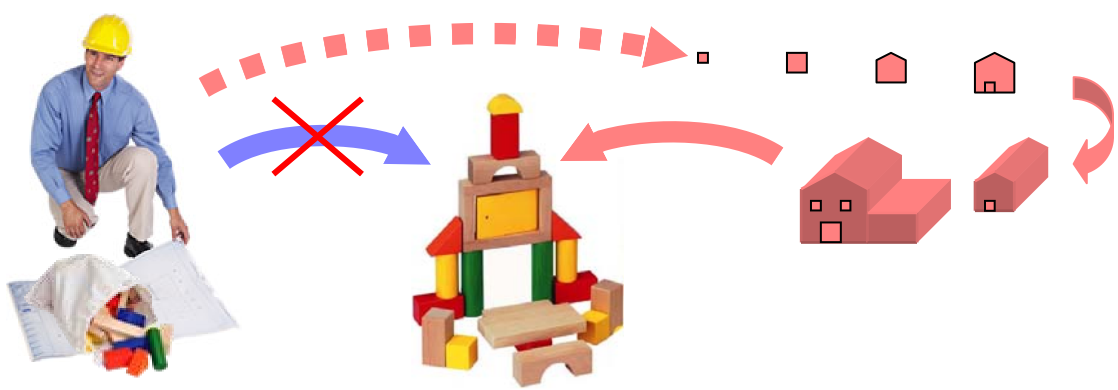
- ... and of evolutionary computing: Toward *“meta-design”*
  - ✓ organisms endogenously *grow* but artificial systems *are built* exogenously





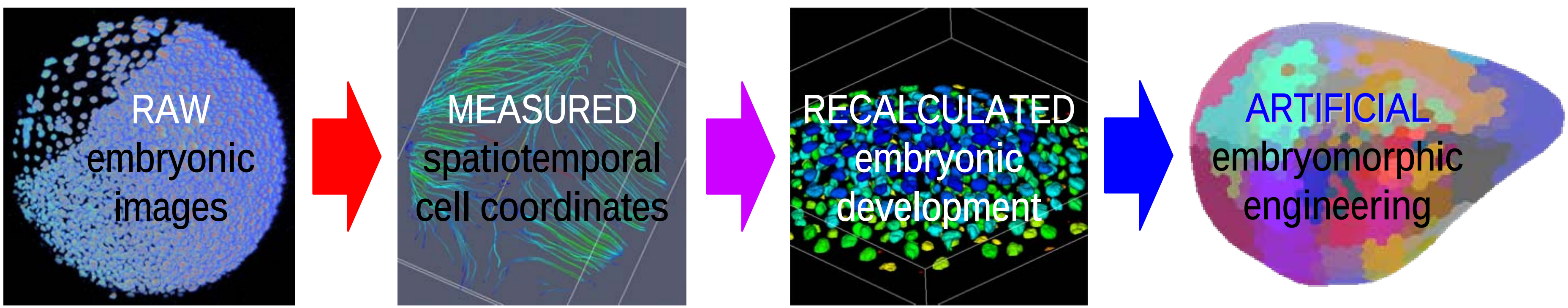
  - ✓ future engineers should “step back” from their creation and only set *generic* conditions for systems to self-assemble and evolve

don’t build the system (phenotype), <u>program the agents</u> (developmental genotype)—see, e.g., “artificial embryogeny”



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*



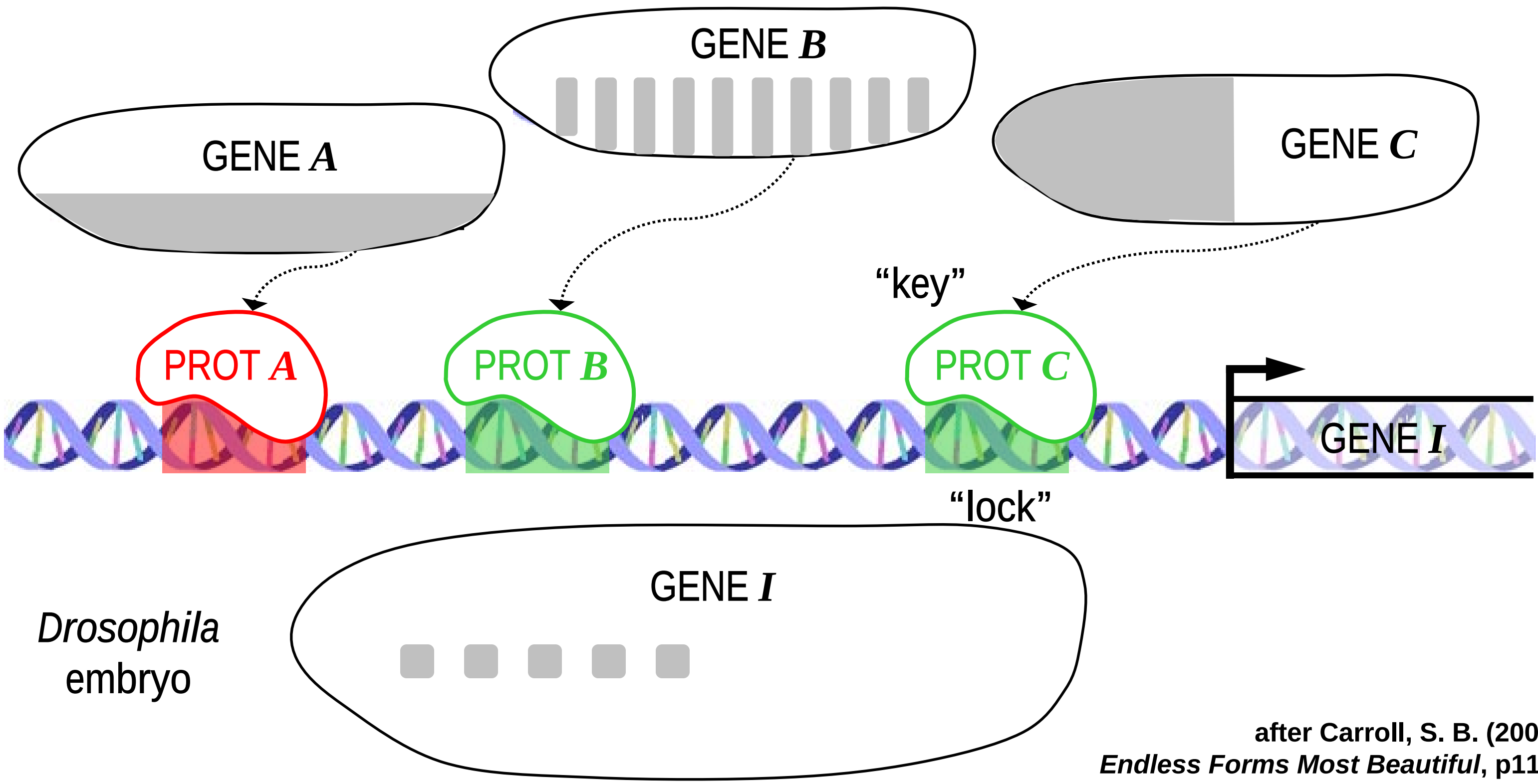
European projects “Embryomics” & “BioEmergences”

Toward Morphogenetic Engineering

1. Self-organized *and* structured complex systems
2. Toward “evo-devo” engineering
3. A model of programmable morphogenesis
4. Evolutionary meta-design
5. Extension: programmable complex networks

3. Programmable morphogenesis

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

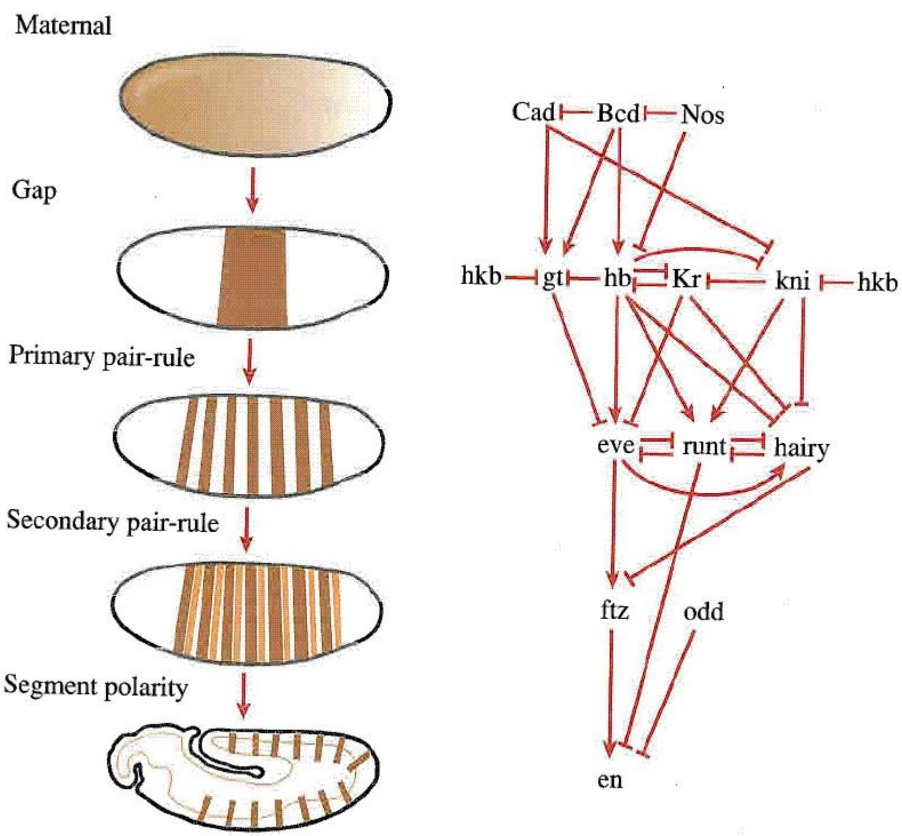


after Carroll, S. B. (2005)
***Endless Forms Most Beautiful*, p117**

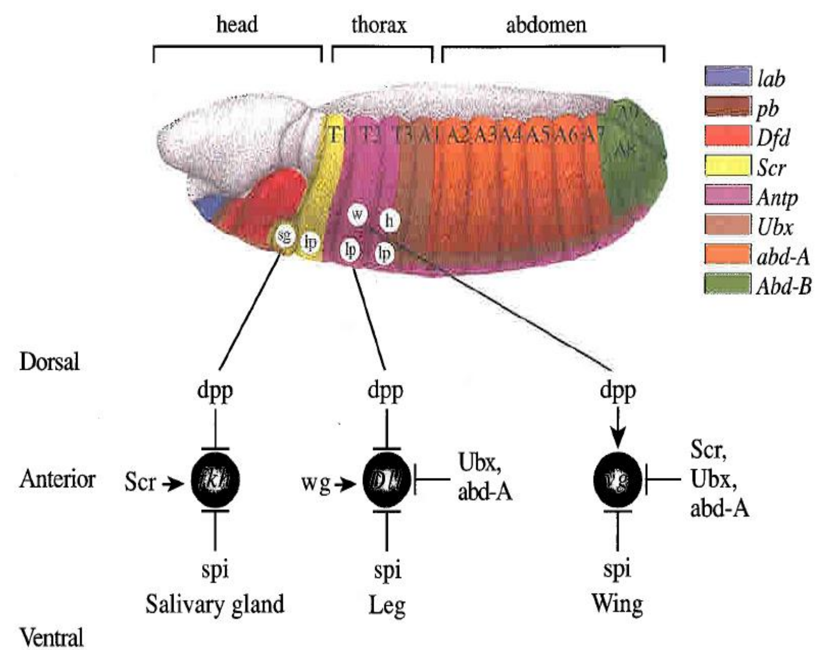
3. Programmable morphogenesis

➢ Segmentation & identity domains in *Drosophila*

✓ periodic A/P band patterns are controlled by a 5-tier gene regulatory hierarchy



✓ intersection with other axes creates organ primordia and imaginal discs (identity domains of future legs, wings, antennae, etc.)

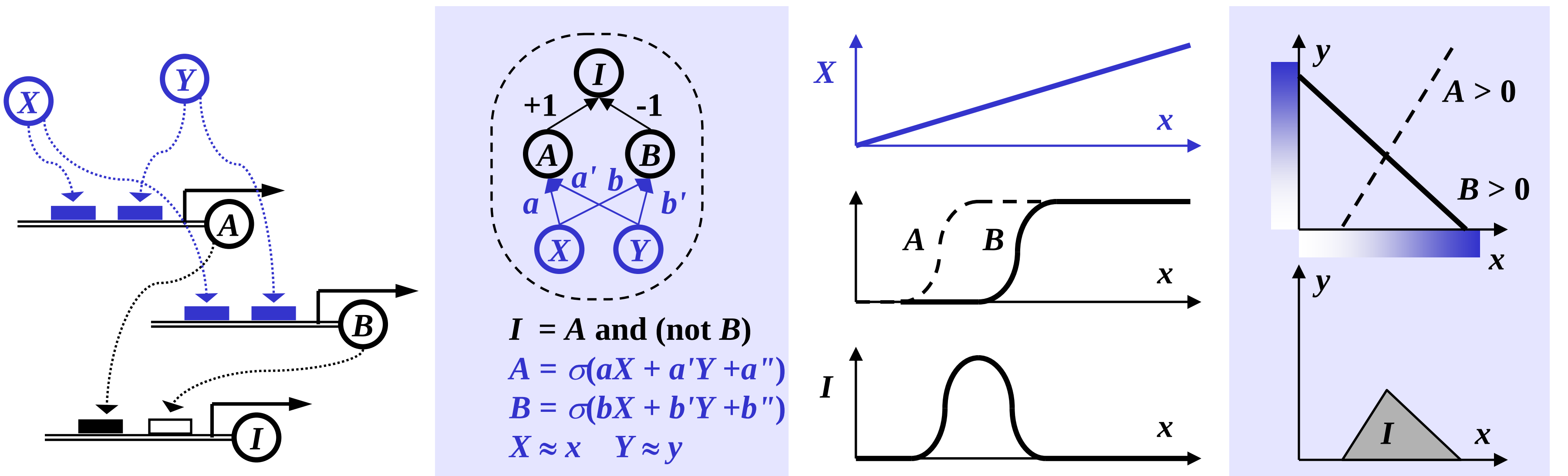


from Carroll, S. B., et al. (2001) *From DNA to Diversity*, p63

3. Programmable 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*

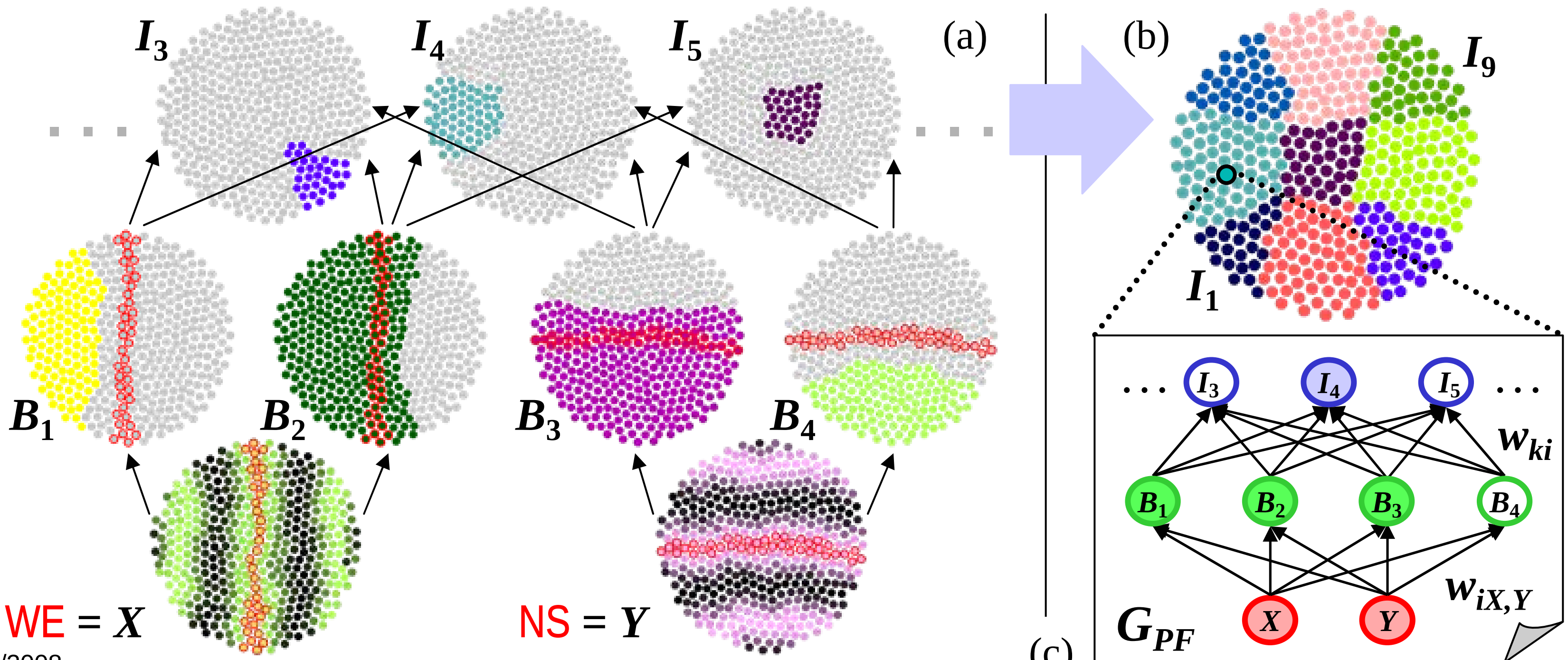
3. Programmable morphogenesis

➢ Programmed patterning (PF-II): the hidden embryo map

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.



3. Programmable morphogenesis

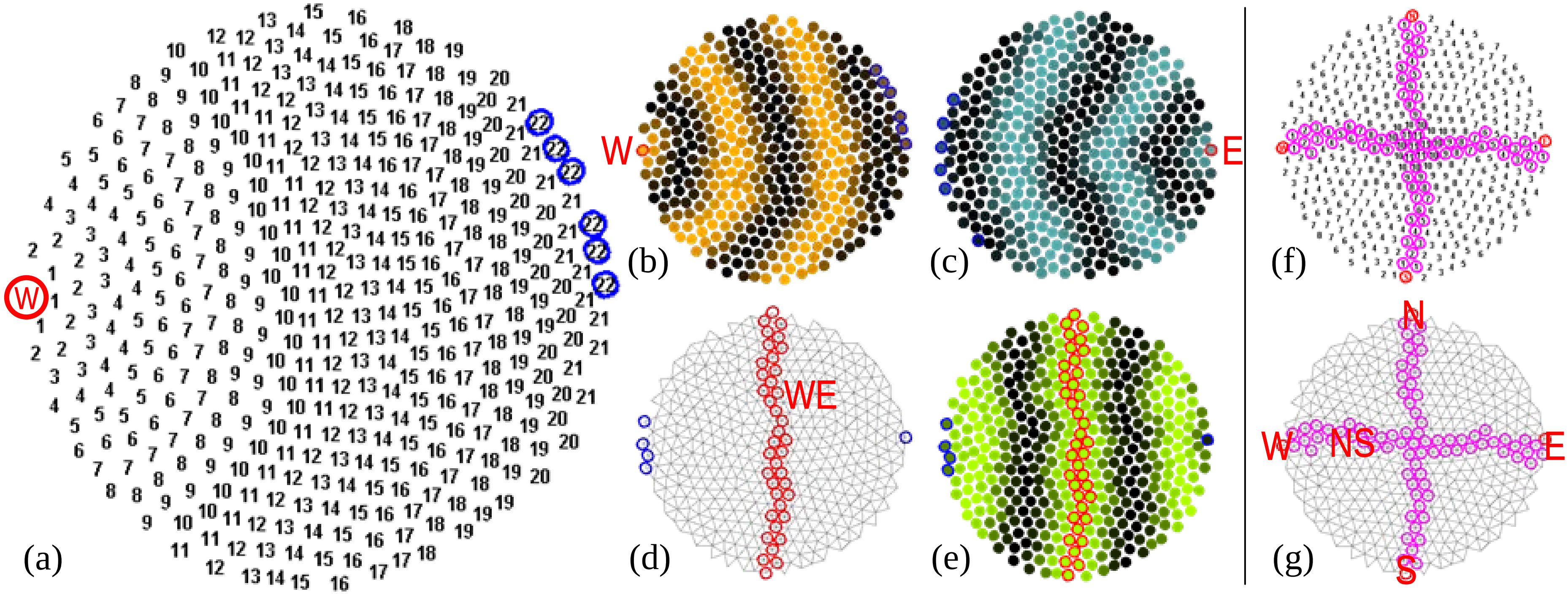
➢ Propagation of positional information (PF-I)

a) & b) circular gradient of counter values originating from source agent *W*

c) opposite gradient coming from antipode agent *E*

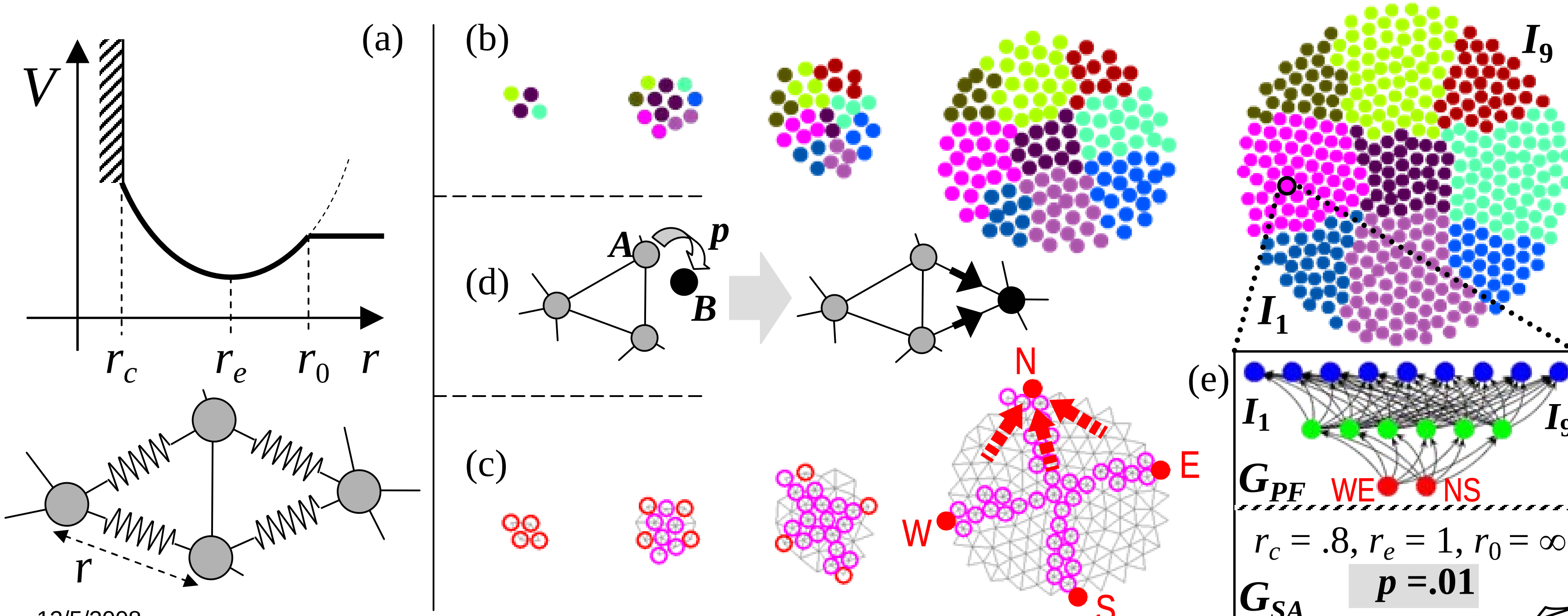
d) & e) planar gradient from *WE* agents (whose *W* and *E* counters equate ±1)

f) & g) complete coordinate compass, with *NS* midline.



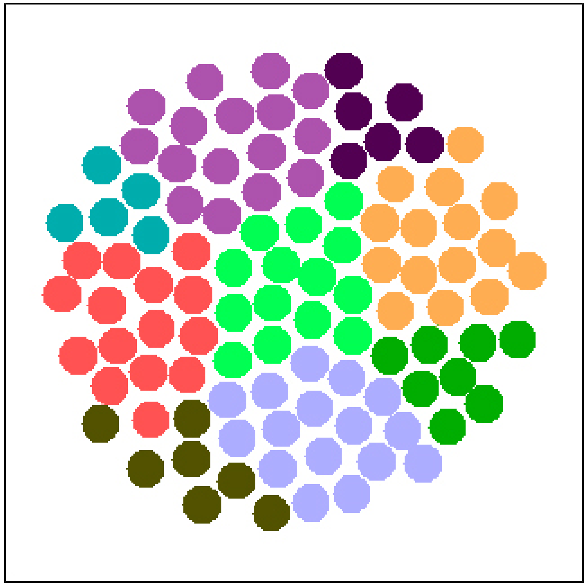
3. Programmable morphogenesis

- Simultaneous growth *and* patterning (SA + PF)
  - a) elastic adhesion forces; b) swarm growing from 4 to 400 agents by division
  - c) swarm mesh, gradient midlines; pattern is continually maintained by source migration, e.g., N moves away from S and toward WE
  - d) agent B created by A's division quickly submits to SA forces and PF traffic
  - e) combined genetic programs inside each agent

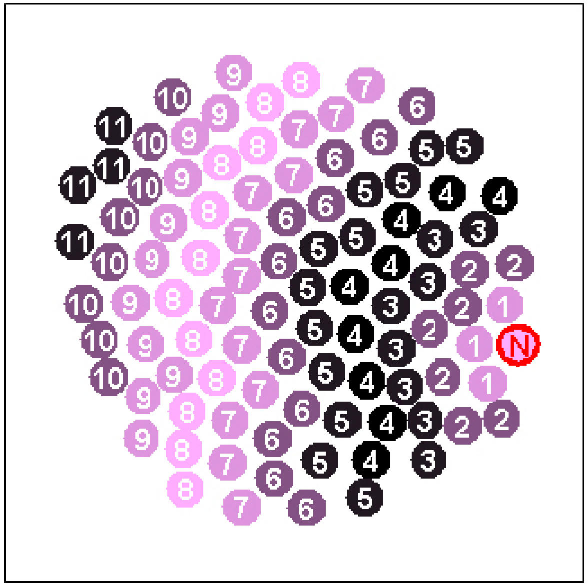


3. Programmable morphogenesis

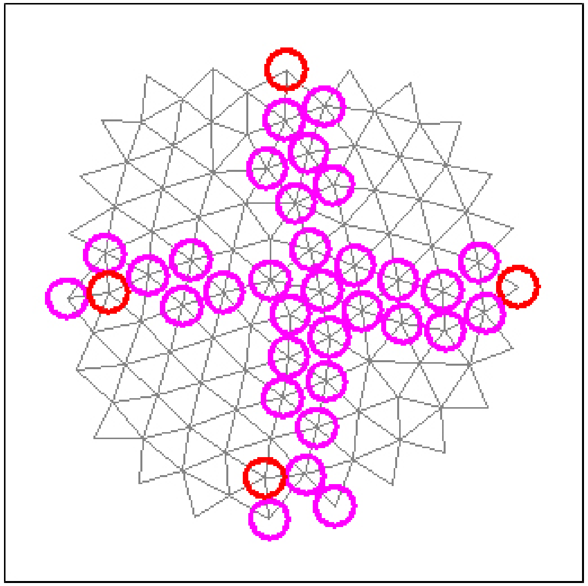
- Simultaneous growth *and* patterning (SA + PF)
  - ✓ example of simulation: 3 movies showing the same development highlighting 3 different planes (in different embryos)



highlighting
gene patterning (PF-II)



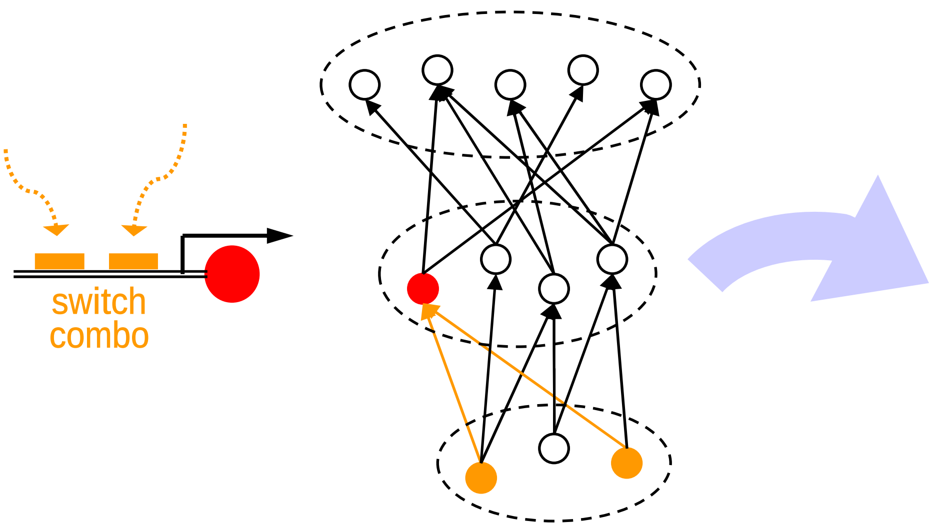
highlighting
gradient formation (PF-I)

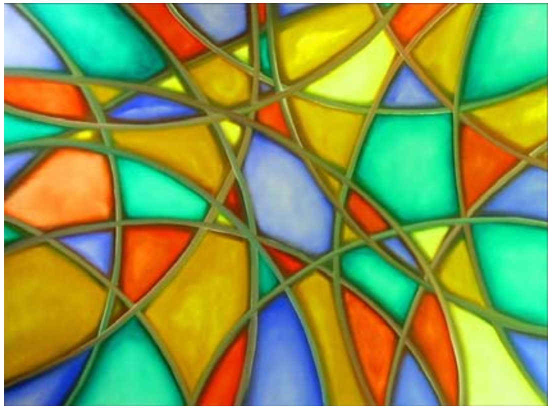


highlighting
lattice (SA) with gradient lines

3. Programmable morphogenesis

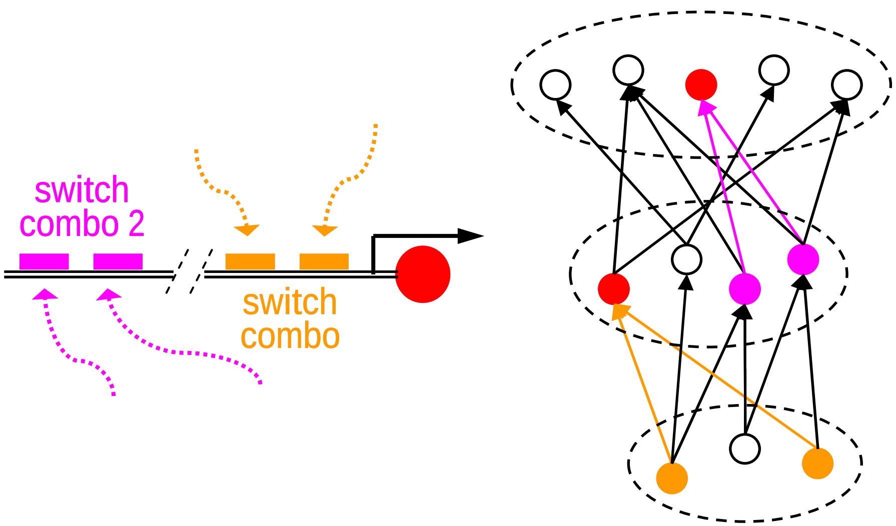
- Summary: simple feedforward hypothesis
  - ✓ developmental genes are broadly organized in tiers, or "generations": earlier genes map the way for later genes
  - ✓ gene expression propagates in a directed fashion: first, positional morphogens create domains, then domains intersect

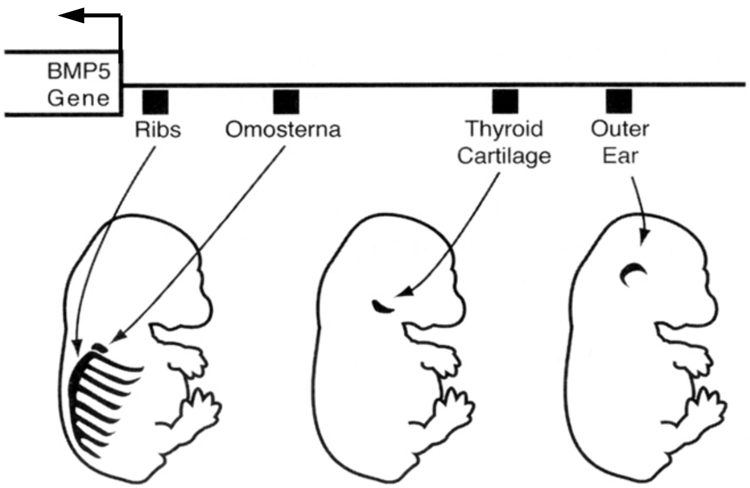




3. Programmable morphogenesis

- Naturally, toolkit genes are often multivalent
  - ✓ exception to the feedforward paradigm: “toolkit” genes that are reused at different stages and different places in the organism
  - ✓ however, a toolkit gene is triggered by different switch combos, which can be represented by duplicate nodes in different tiers



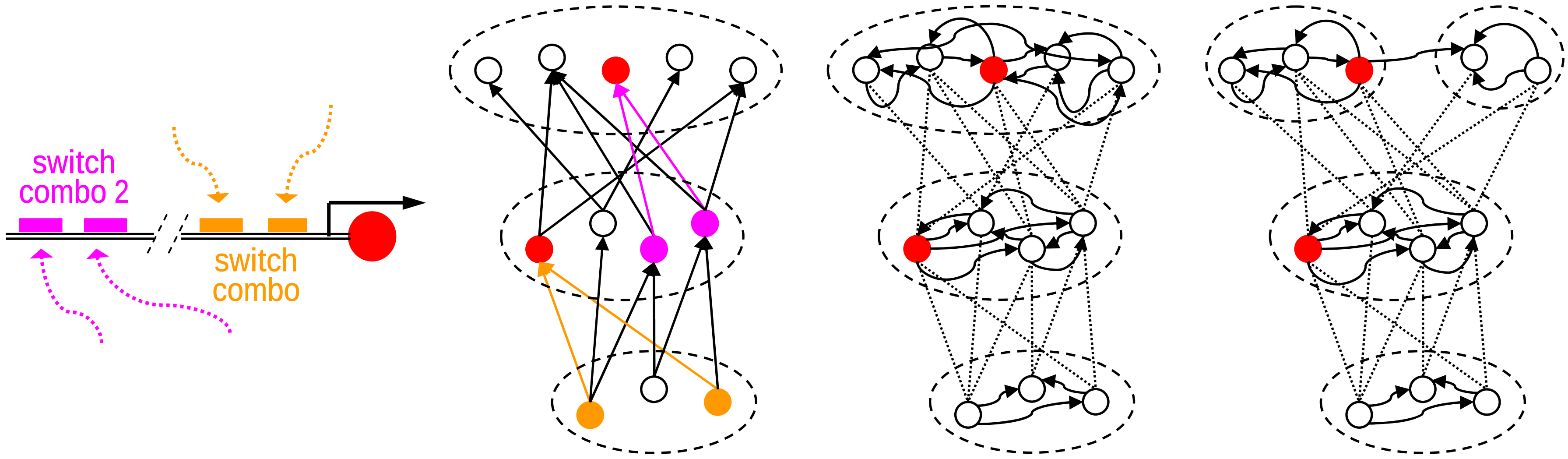


after David Kingsley, in Carroll, S. B. (2005)
Endless Forms Most Beautiful, p125

3. Programmable morphogenesis

➢ More realistic variants of GRNs

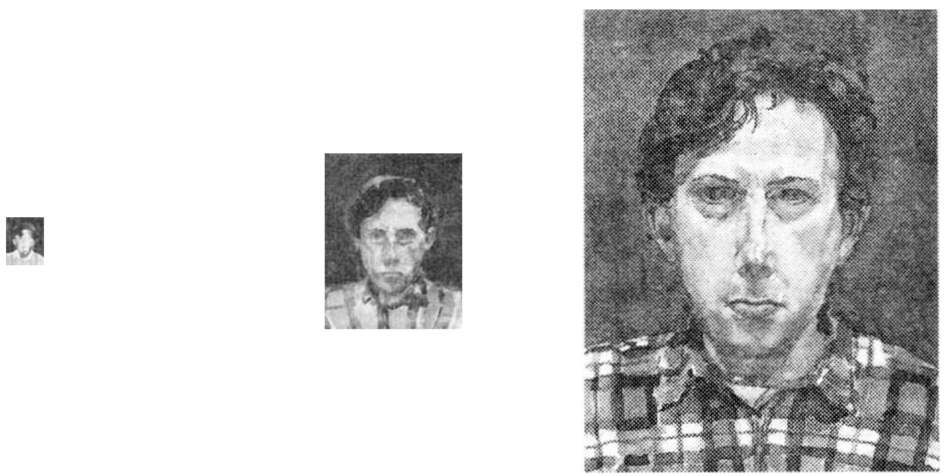
- ✓ add recurrent links within tiers → domains are not established independently but influence and sharpen each other
- ✓ subdivide tiers into subnetworks → this creates **modules** that can be reused and starts a hierarchical architecture



3. Programmable morphogenesis

- Morphological refinement by iterative growth
  - ✓ details are not created in one shot, but gradually added. . .

  - ✓ . . . while, at the same time, the canvas grows



from Coen, E. (2000)
The Art of Genes, pp131-135

3. Programmable morphogenesis

➢ Modular, recursive patterning (PF[k])...

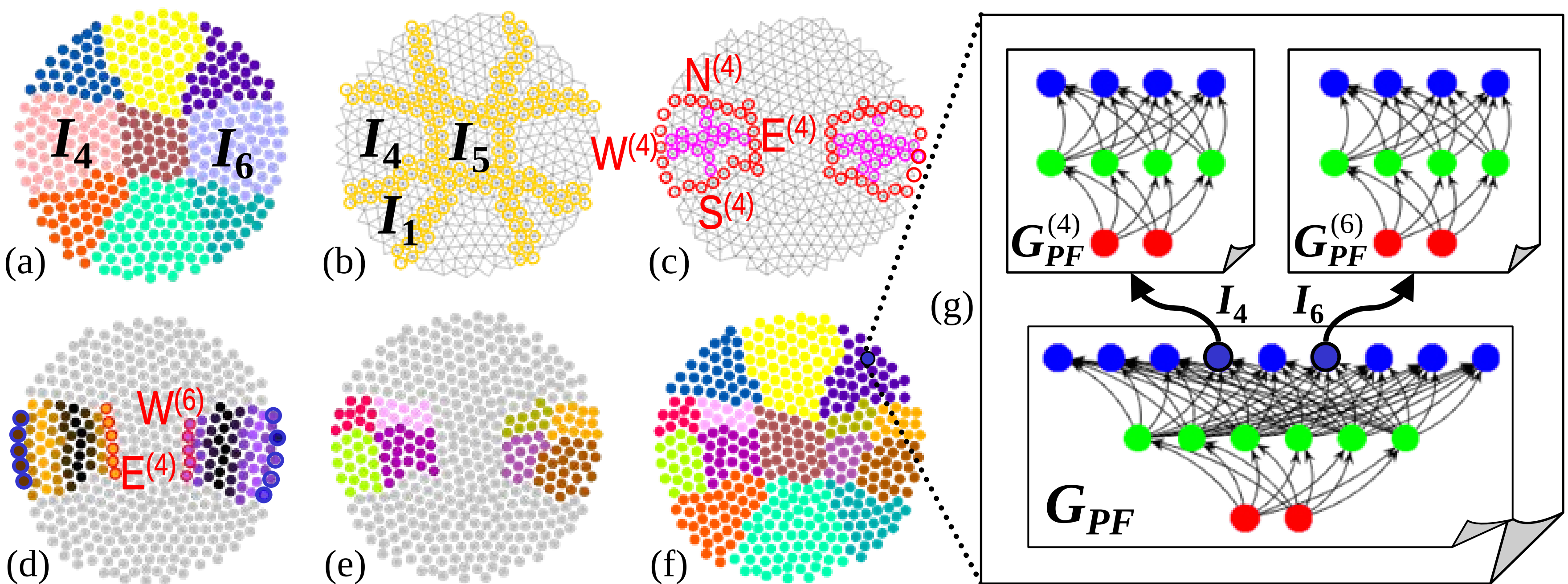
b) border agents highlighted in yellow

c) border agents become new gradient sources inside certain identity regions

d) missing border sources arise from the ends (blue circles) of other gradients

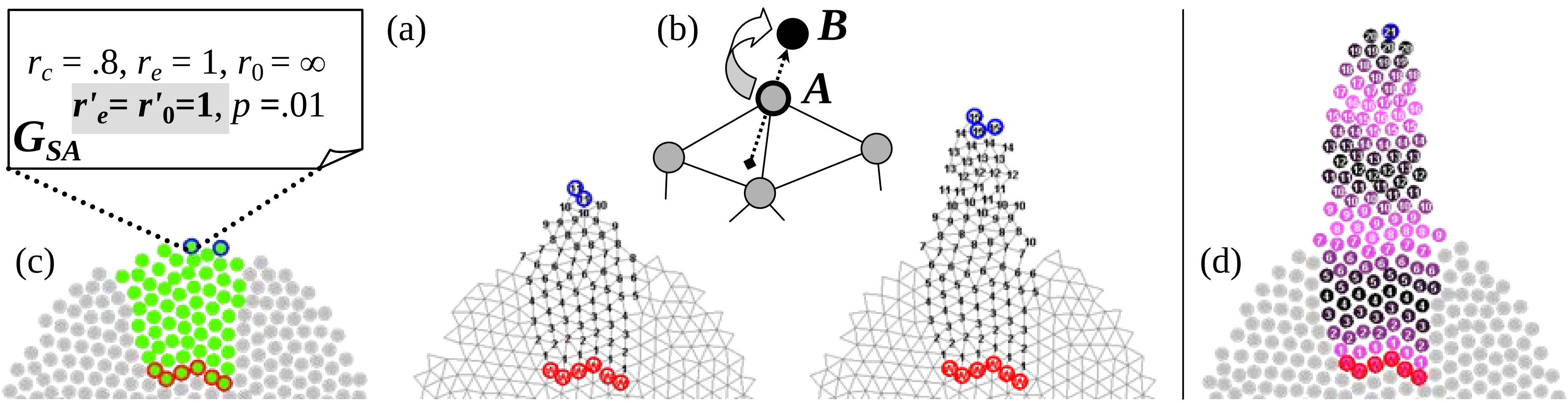
e) & f) subpatterning of the swarm in I_4 and I_6

g) corresponding hierarchical gene regulation network



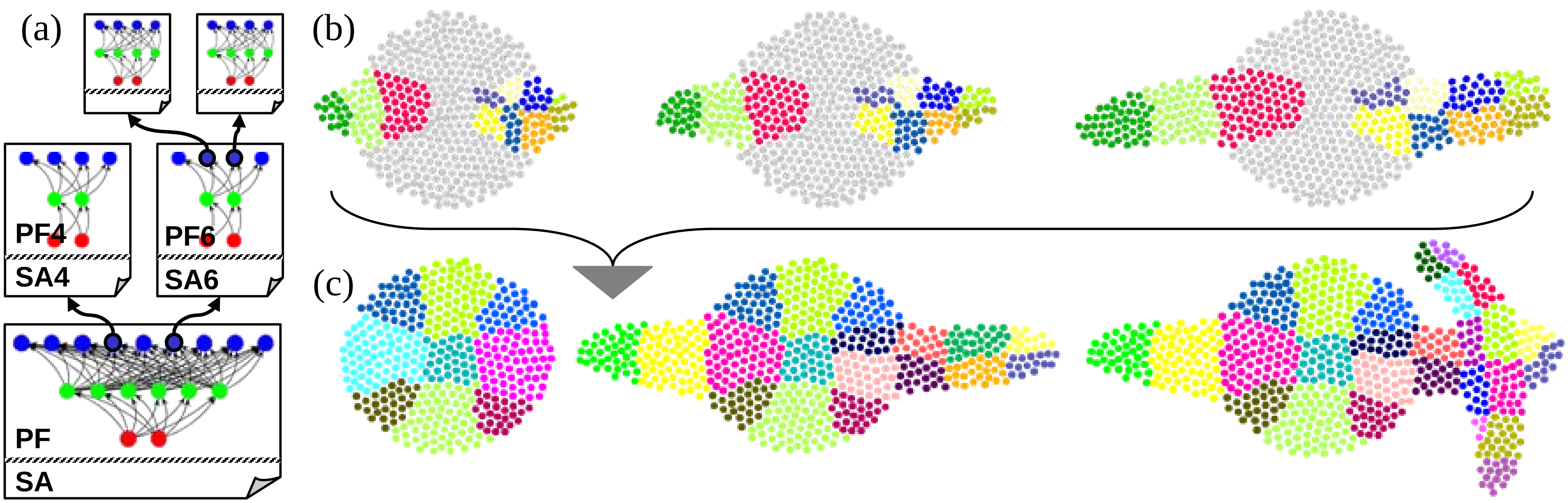
3. Programmable morphogenesis

- ... in parallel with modular, anisotropic growth (SA[k])
  - a) genetic SA parameters are augmented with repelling V values r'_e and r'_0 used between the growing region (green) and the rest of the swarm (gray)
  - b) daughter agents are positioned away from the neighbors' center of mass
  - c) offshoot growth proceeds from an "apical meristem" made of gradient ends (blue circles)
  - d) the gradient underlying this growth



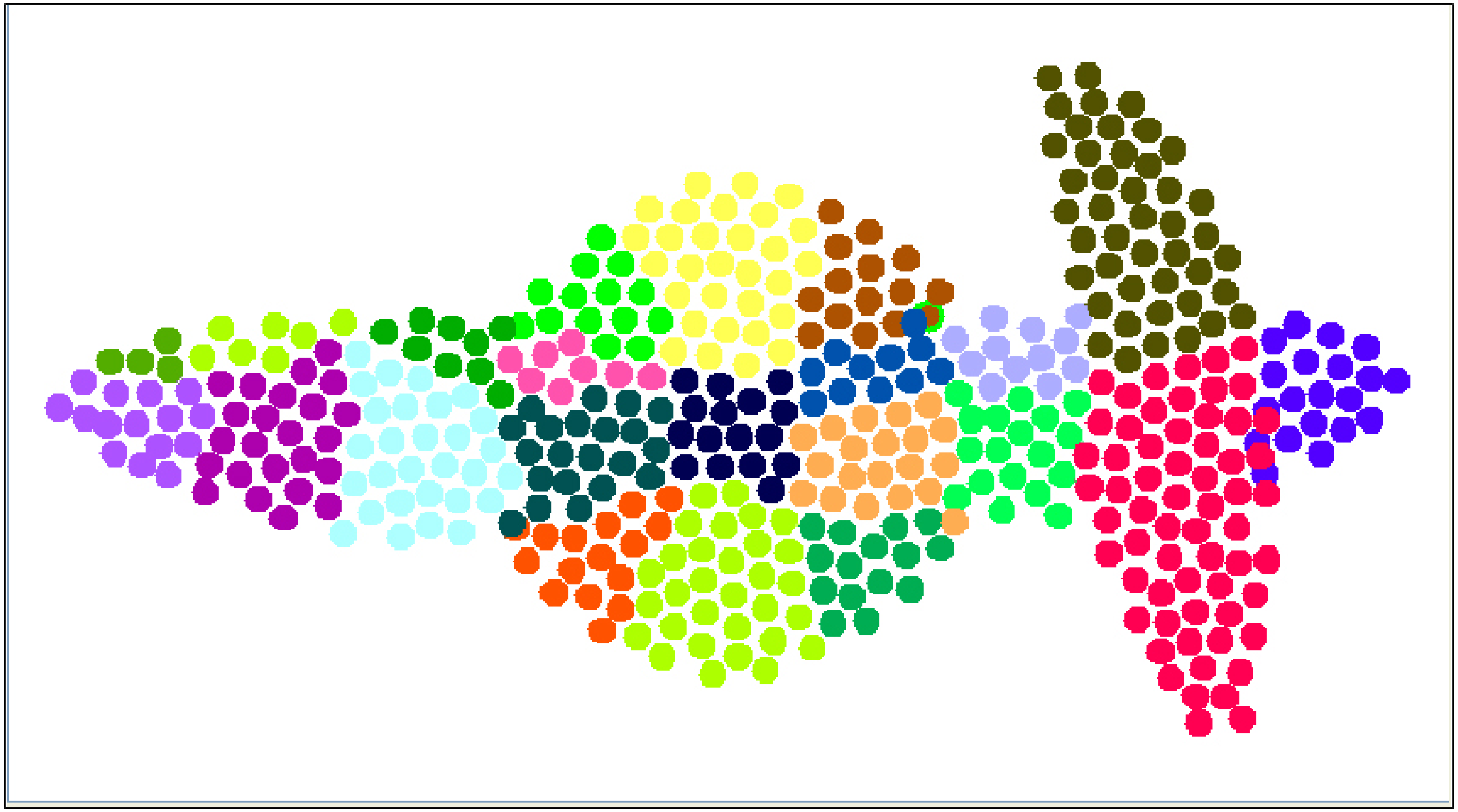
3. Programmable morphogenesis

- Modular growth *and* patterning (SA[k] + PF[k]): 3 levels
  a) example of a three-level modular genotype giving rise to the artificial organism on the right
  b) three iterations detailing the simultaneous limb-like growth process and patterning of these limbs during execution of level 2 (modules 4 and 6)
  c) main stages of the complex morphogenesis, showing full patterns after execution of levels 1, 2 and 3.



3. Programmable morphogenesis

- Modular growth *and* patterning (SA[k] + PF[k]): 3 levels

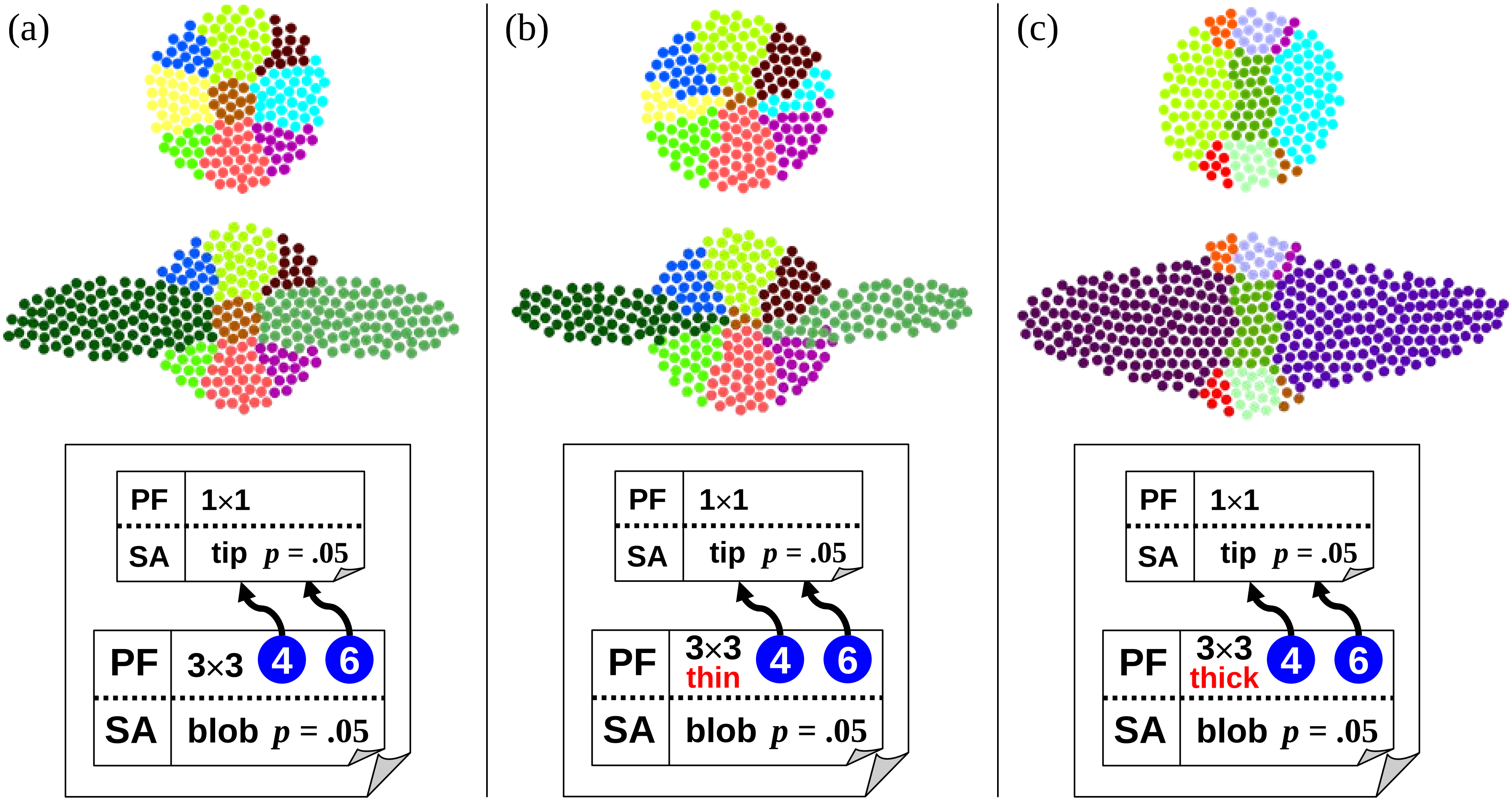


Toward Morphogenetic Engineering

1. Self-organized *and* structured complex systems
2. Toward “evo-devo” engineering
3. A model of programmable morphogenesis
4. Evolutionary meta-design
5. Extension: programmable complex networks

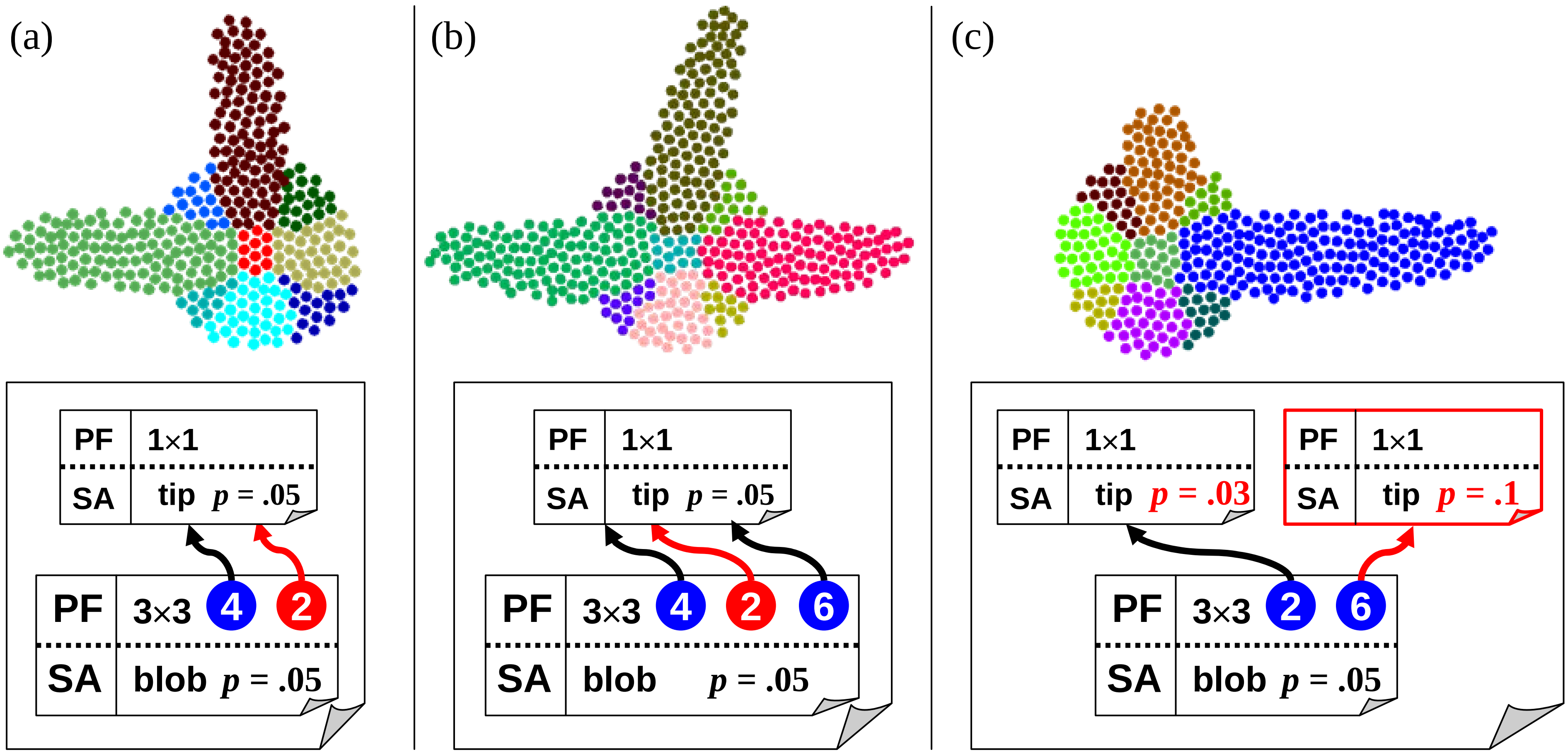
4. Evolutionary meta-design

- Modular growth and patterning (SA[k] + PF[k]): 2 levels
  a) wild type; b) “thin” mutation of the base body plan; c) “thick” mutation



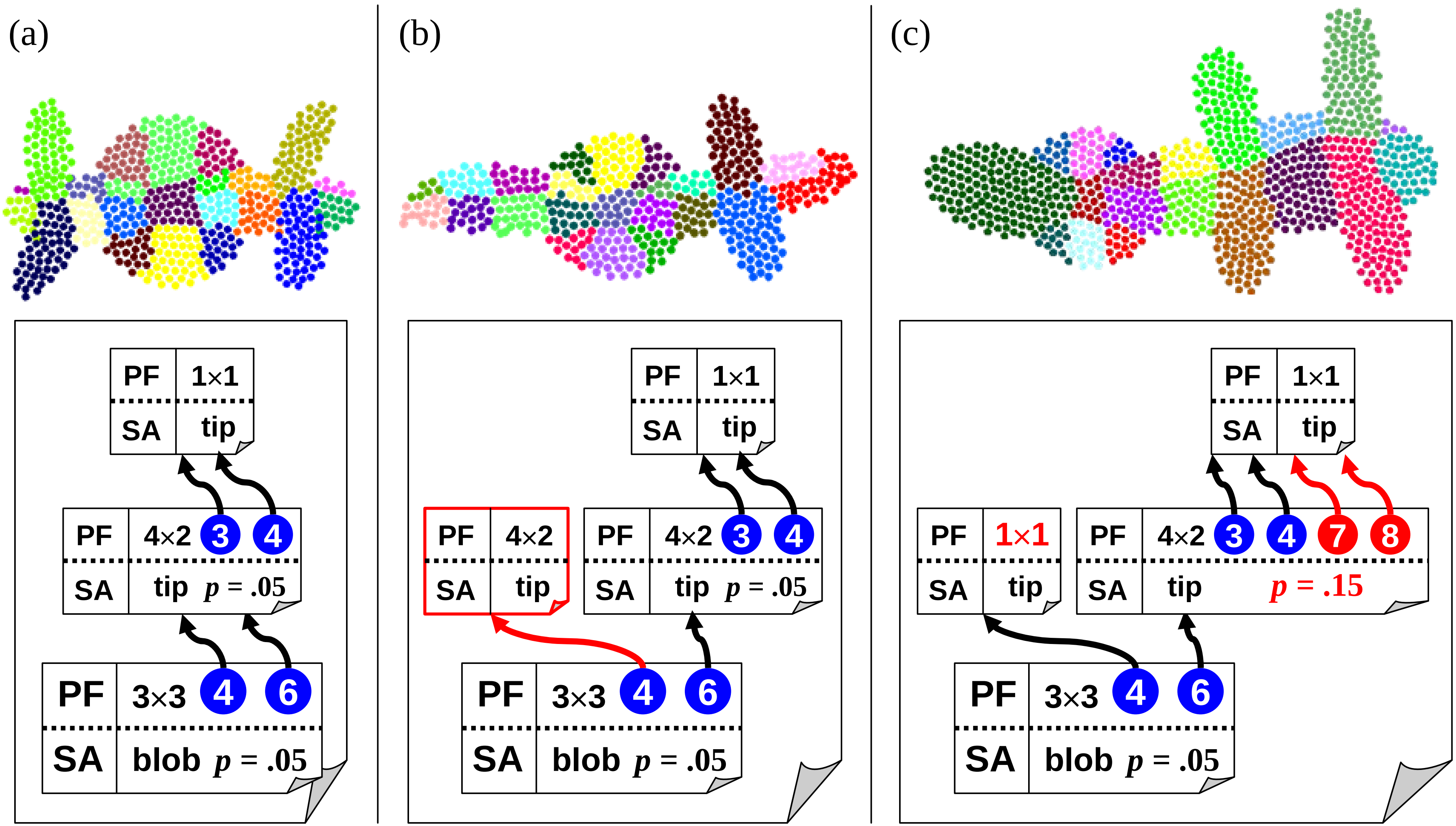
4. Evolutionary meta-design

- ➢ **Modular growth and patterning (SA[k] + PF[k]): 2 levels**
  - a) antennapedia; b) homology by *duplication*; c) *divergence* of the homology



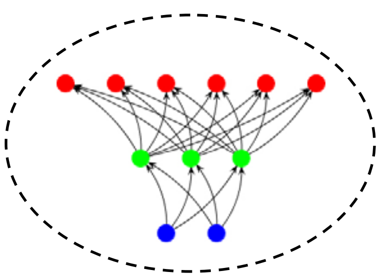
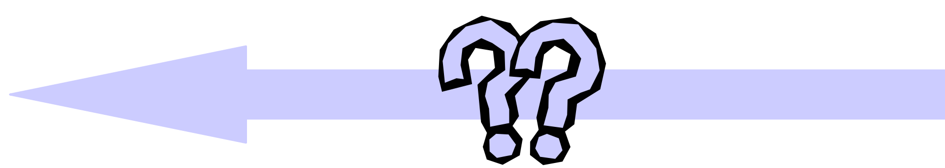
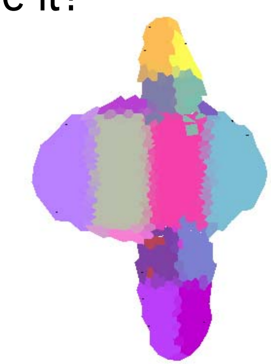
4. Evolutionary meta-design

- Modular growth and patterning (SA[k] + PF[k]): 3 levels



4. Evolutionary Meta-Design

- 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?

4. Evolutionary Meta-Design

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

parameters = "genetic code"

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.

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

4. Evolutionary Meta-Design

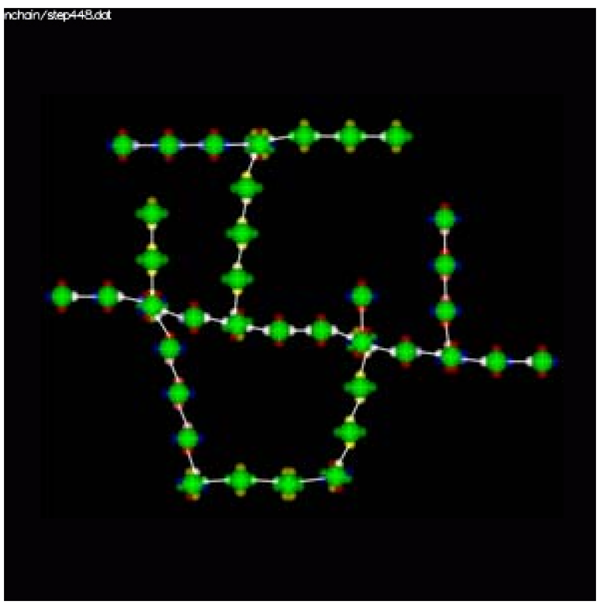
- Selecting without expectations?
  - ✓ different degrees of fitness constraints
  - a) selecting for a specific **organism** (shape, pattern)
    - reverse problem: given the phenotype, what should be the genotype?
    - **direct** recipe; ex: Nagpal's macro-to-microprogram Origami compilation
    - otherwise: **learn** or **evolve** under strict fitness → 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**: open-ended evolution
    - 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

Toward Morphogenetic Engineering

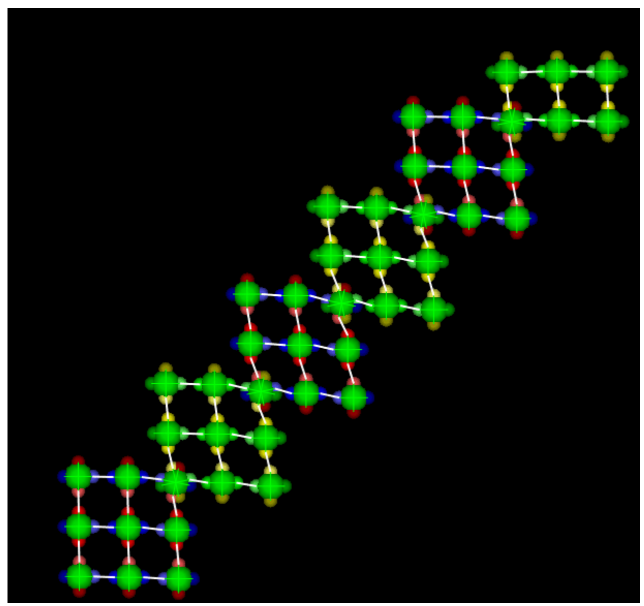
1. Self-organized *and* structured complex systems
2. Toward “evo-devo” engineering
3. A model of programmable morphogenesis
4. Evolutionary meta-design
5. Extension: programmable complex networks

From flocks to shapes

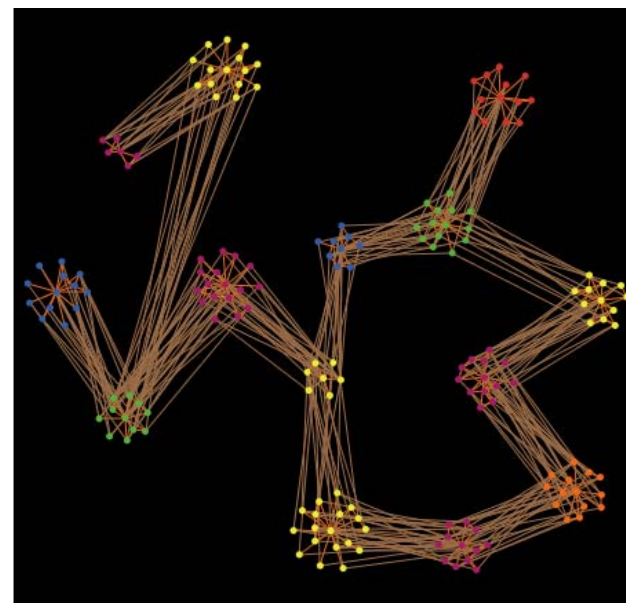
From scale-free to structured networks



single-node
composite branching



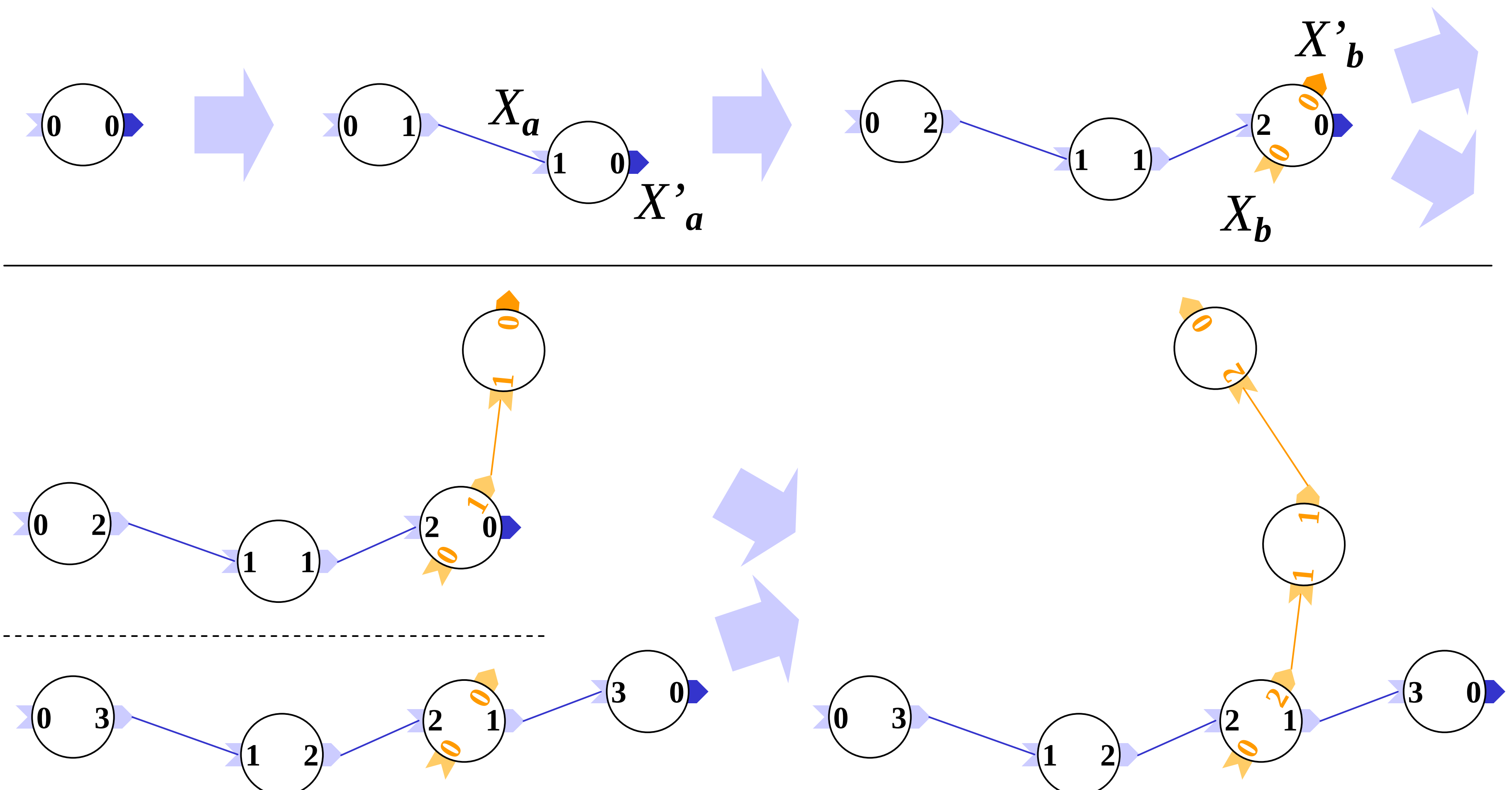
iterative lattice pile-up



clustered
composite branching

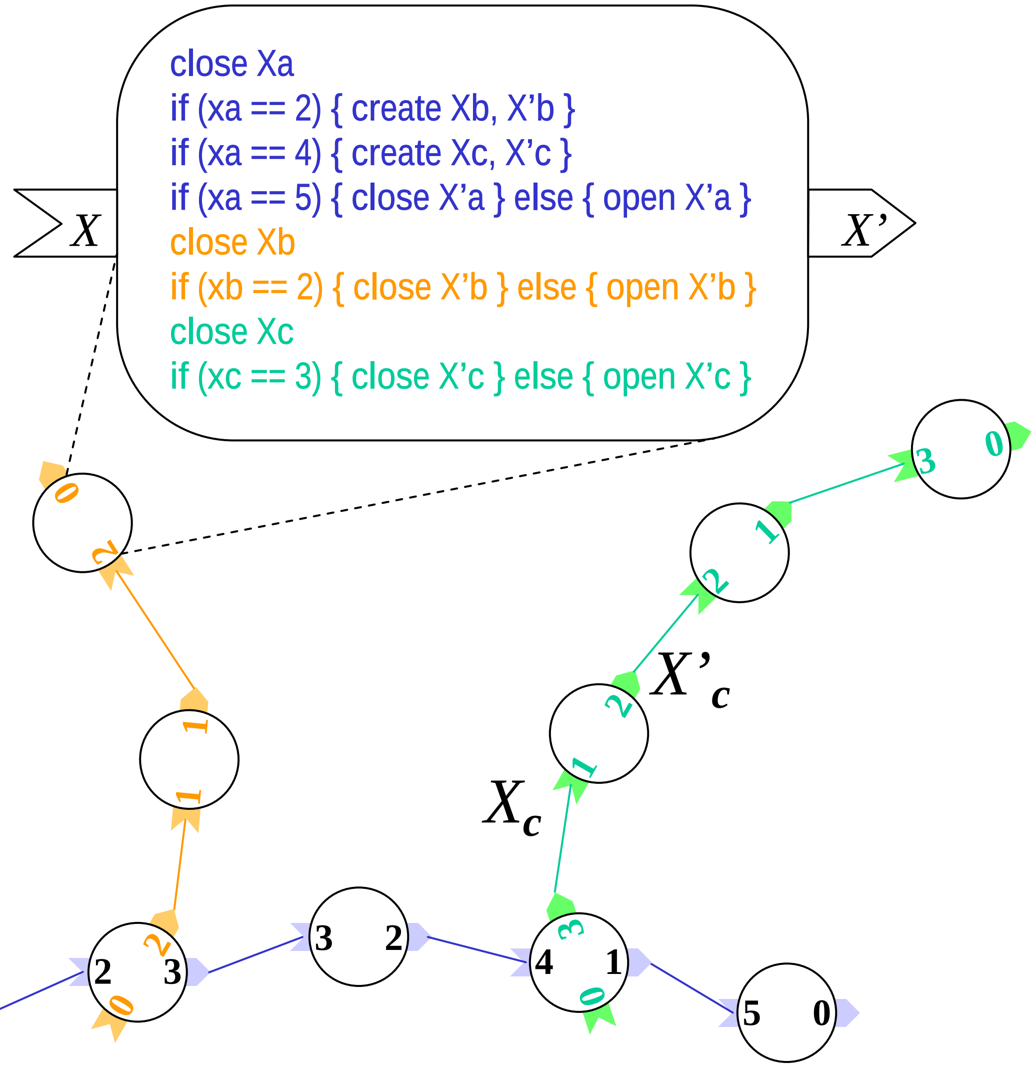
5. Programmable complex networks

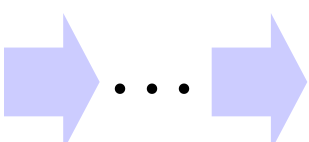
- From preferential to programmed attachment
  - ✓ modular structures by local counters and port logic



5. Programmable complex networks

- From preferential to programmed attachment
  - ✓ the node routines are the *“genotype”* of the network





Morphogenetic Engineering Workshop, Paris 2009

http://www.iscpif.fr/MEW2009

Exporing various engineering approaches to the
artificial design and implementation of autonomous systems capable of
developing complex, heterogeneous morphologies

Thank you