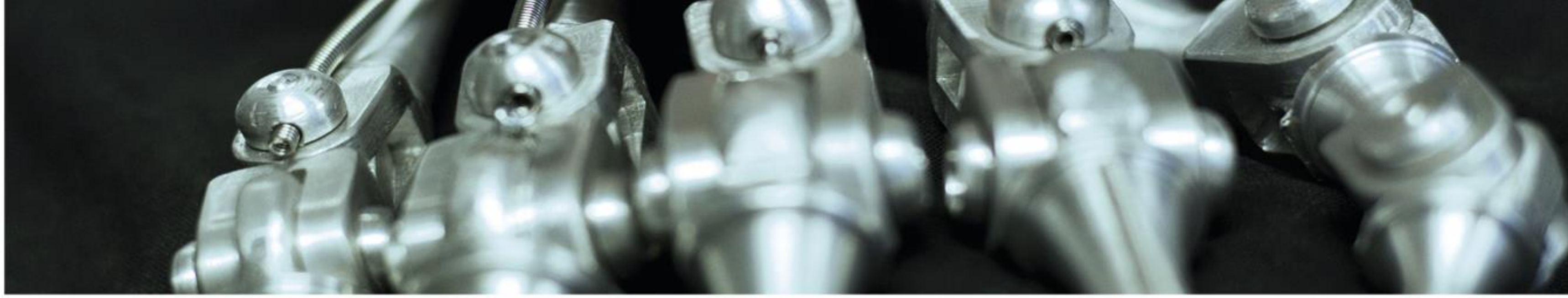


UiO : Department of Informatics
University of Oslo

Frank Veenstra

Swarms and Developmental Systems

Overview

- Biological background
  - Genotype-to-phenotype mapping
- Artificial Developmental Systems
  - Chemistry based approaches (Morphogens)
  - Rewrite systems (L-System)
  - Neural networks (CPPN)
- Swarm Intelligence
  - Particle Swarm Optimization
  - Ant Colony Optimization
  - Reconfigurable Robots

Exercises

- Particle swarm optimization
- Reaction-diffusion model (Gray-scott algorithm)

Analogize to optimize or analogize to understand

Evolution emerges through **reproduction**:

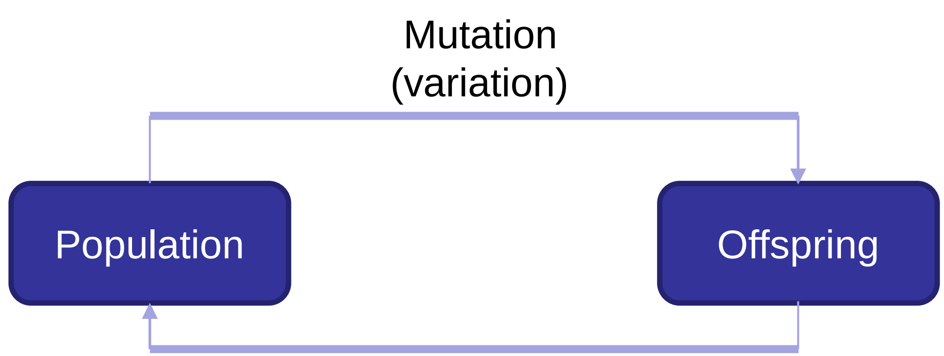
- Storage, processing and transfer of biological information

DNA

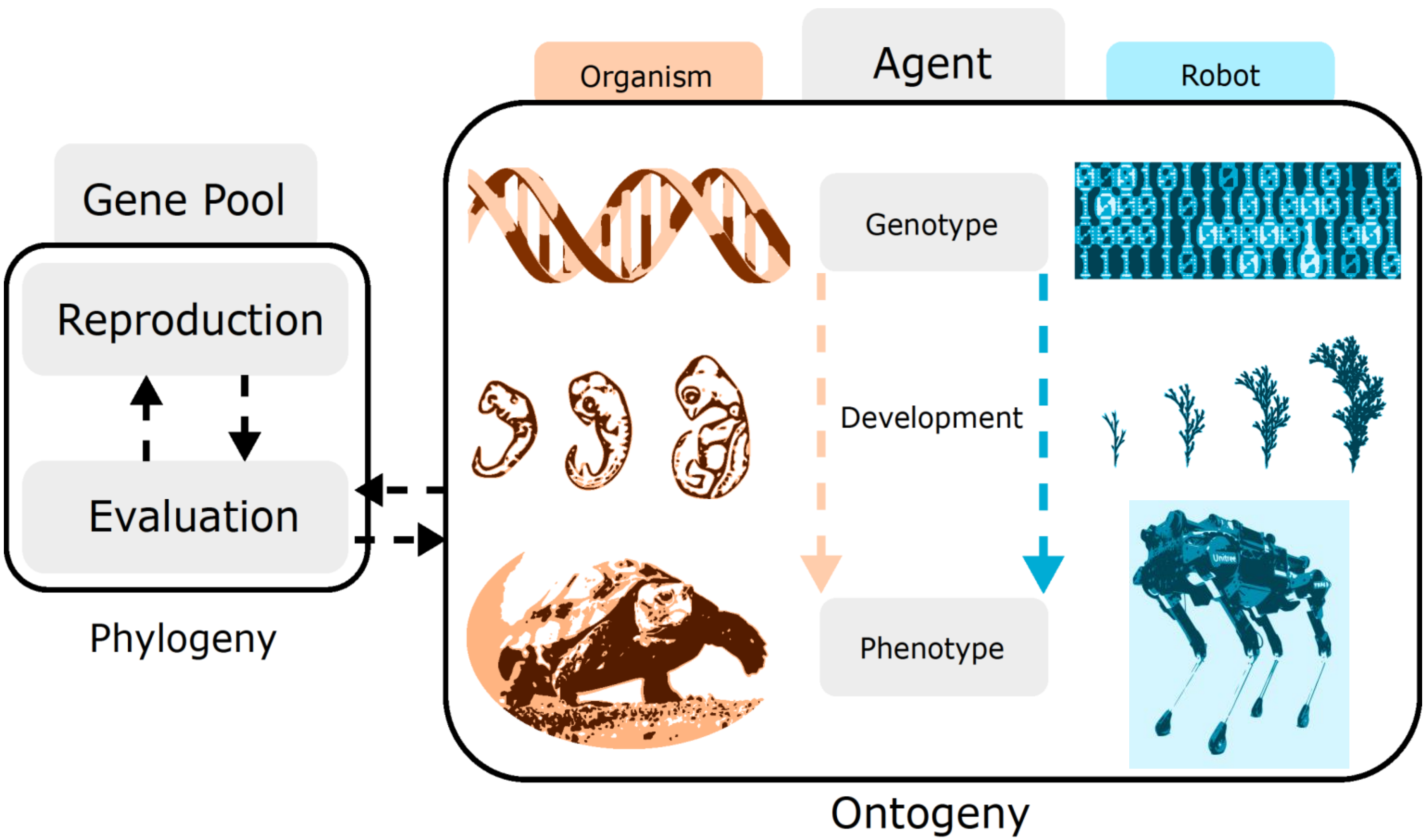
Genotype to phenotype

Offspring

Evolutionary loop



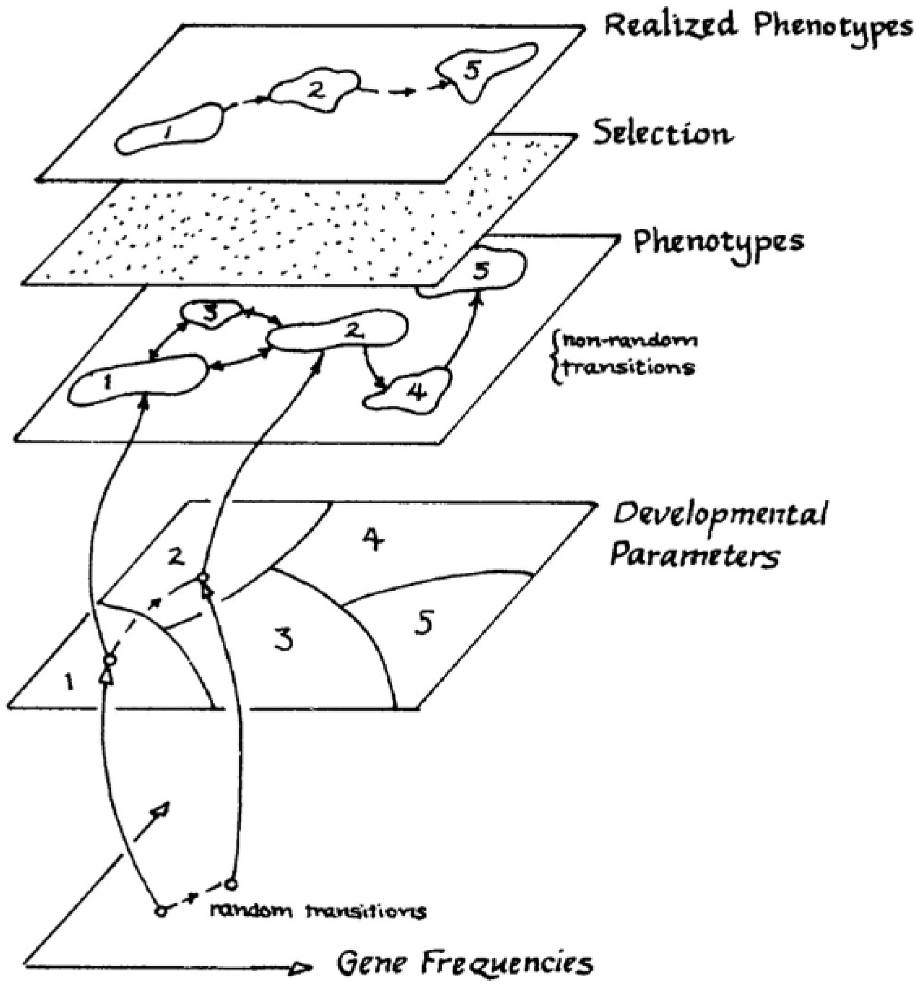
Genotype-to-phenotype mapping

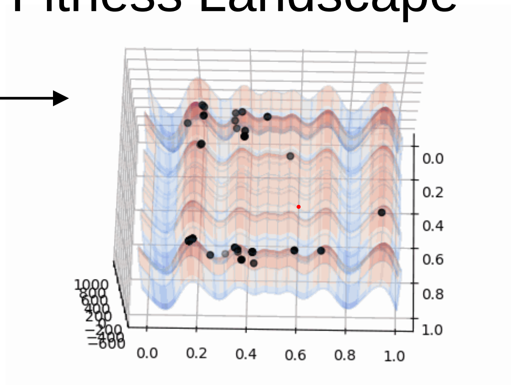


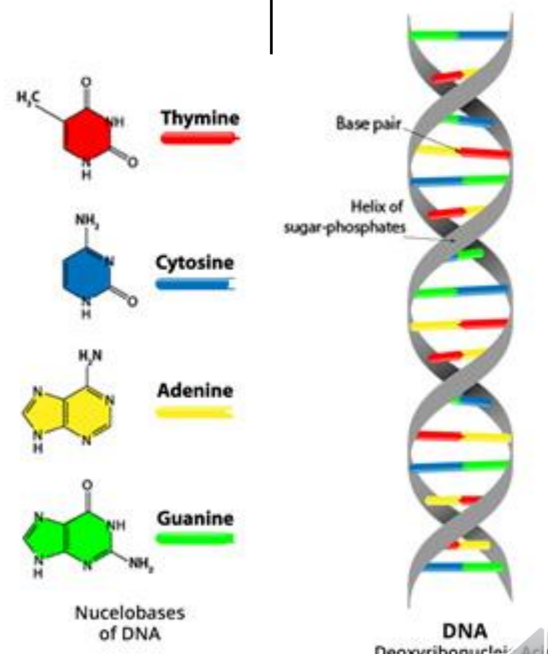
Genotype:
Hereditary

Phenotype:
Observed Properties

Fitness Landscape

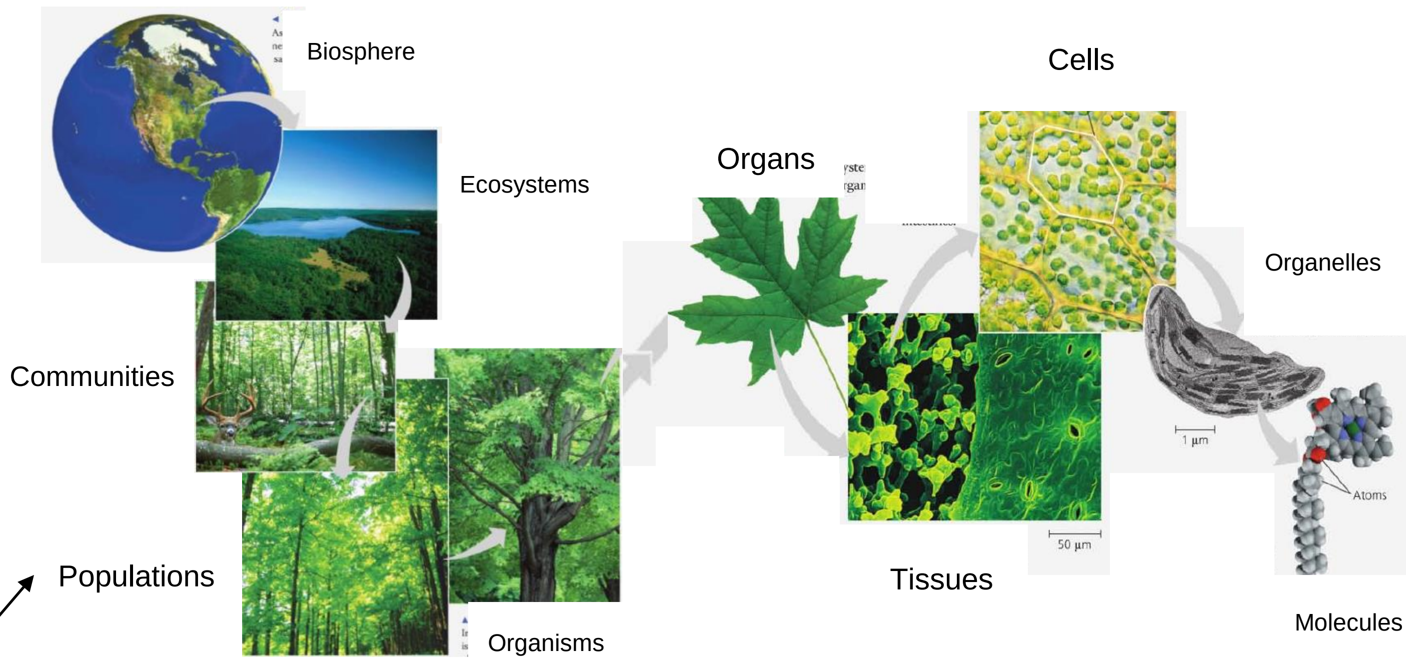






Oster, G. F., & Alberch, P. (1982). Evolution and bifurcation of developmental programs.

Biological Organization



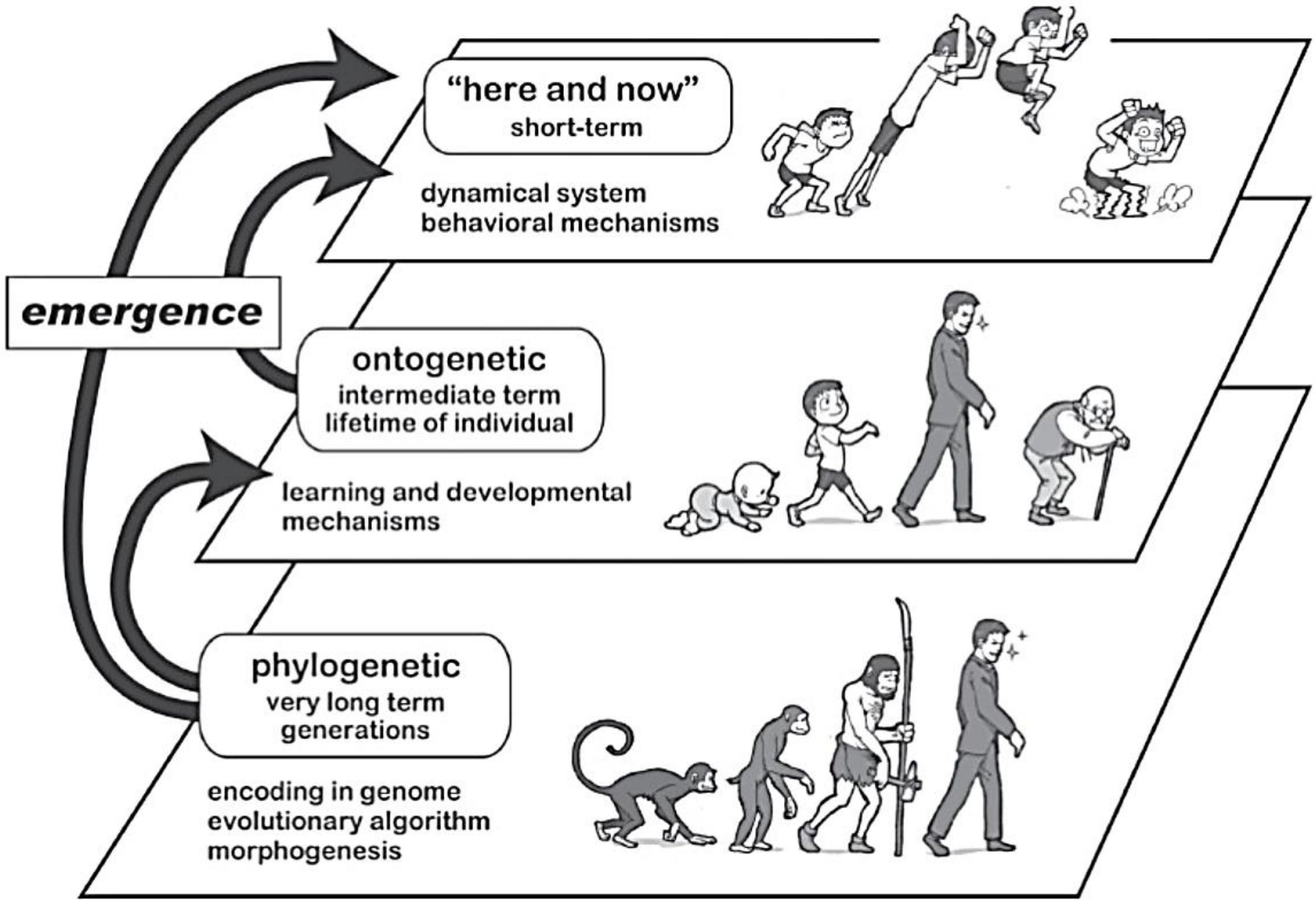
Swarm?

emergence

Phenotype

Campbell and Reece
Biology, 8th ed. 2008

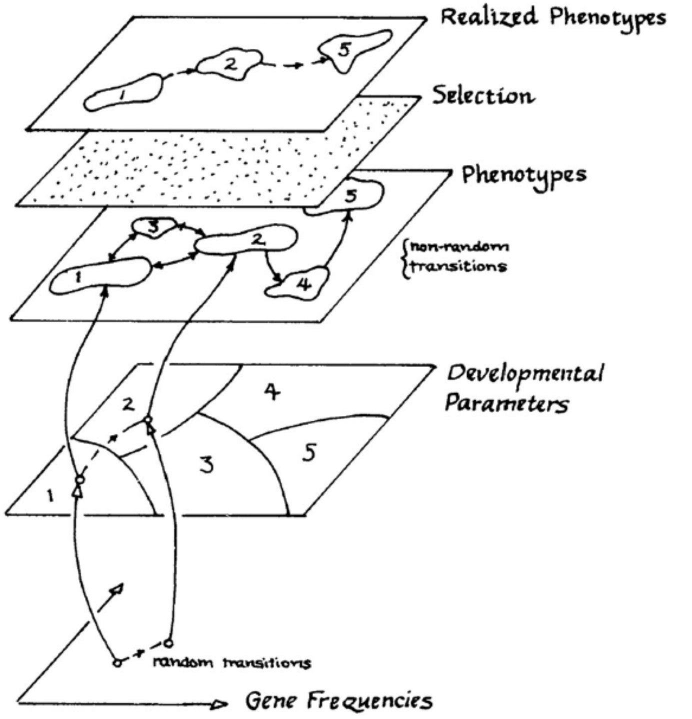
Time scales



Source: Pfeiffer and Bongard (2006) how the body shapes the way we think

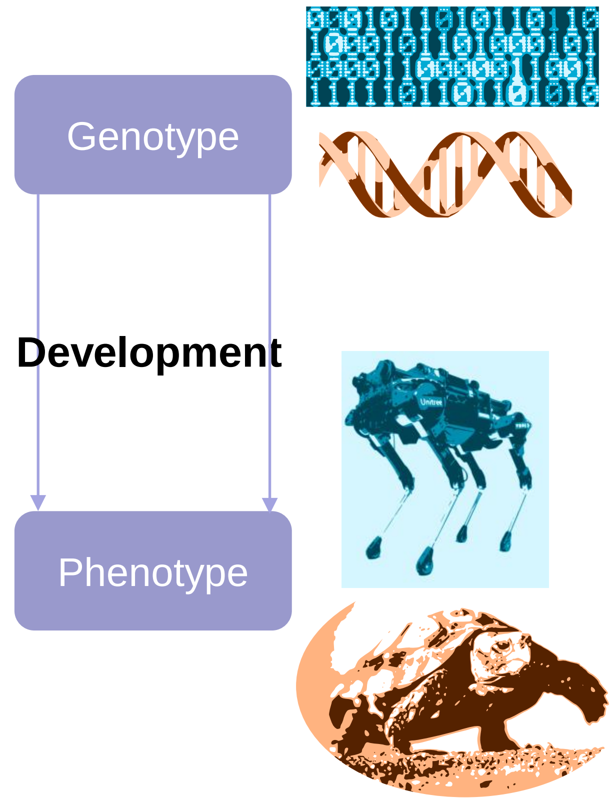
Developmental Systems and Swarms

- Developmental systems act on the genotype to phenotype mapping.
  - Ontogenetic time-scale:
    - In biology from a fertilized egg to a adult organism
- And extended genotype to phenotype map can define collective behaviors seen in swarms



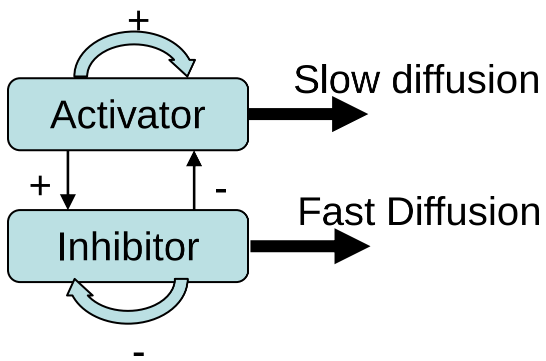
Developmental Systems

- Reuse of information
  - Compact storage
- Modularity
- Self-similarity
- Symmetry
- Scalability



Morphogens

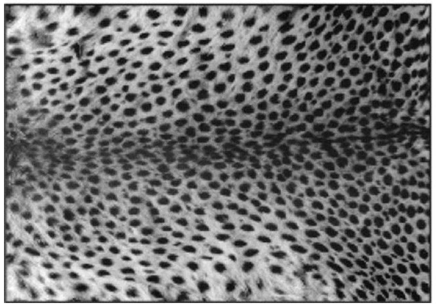
- Morphogenesis:
  - The development of morphological characteristics
- Reaction-diffusion models through self-inhibition

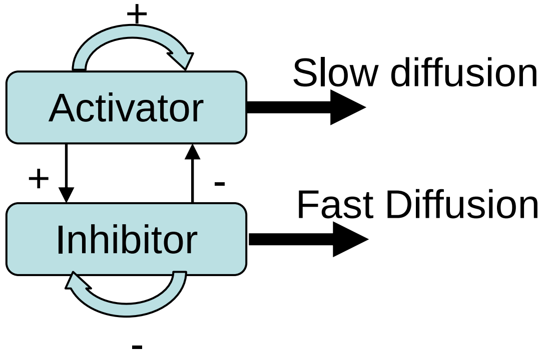


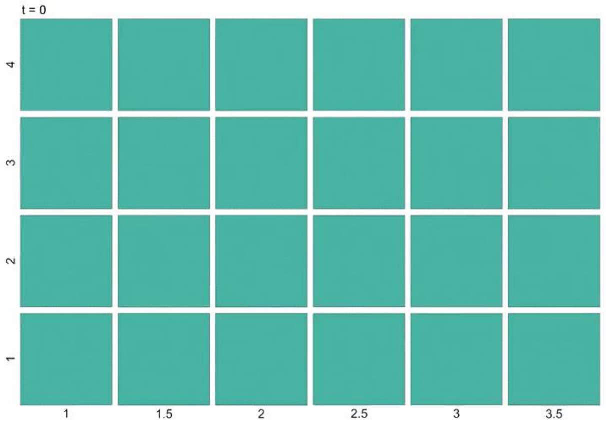
Alan Turing (1952) The Chemical Basis of Morphogenesis

Morphogens

- Reaction-diffusion models through self-inhibition

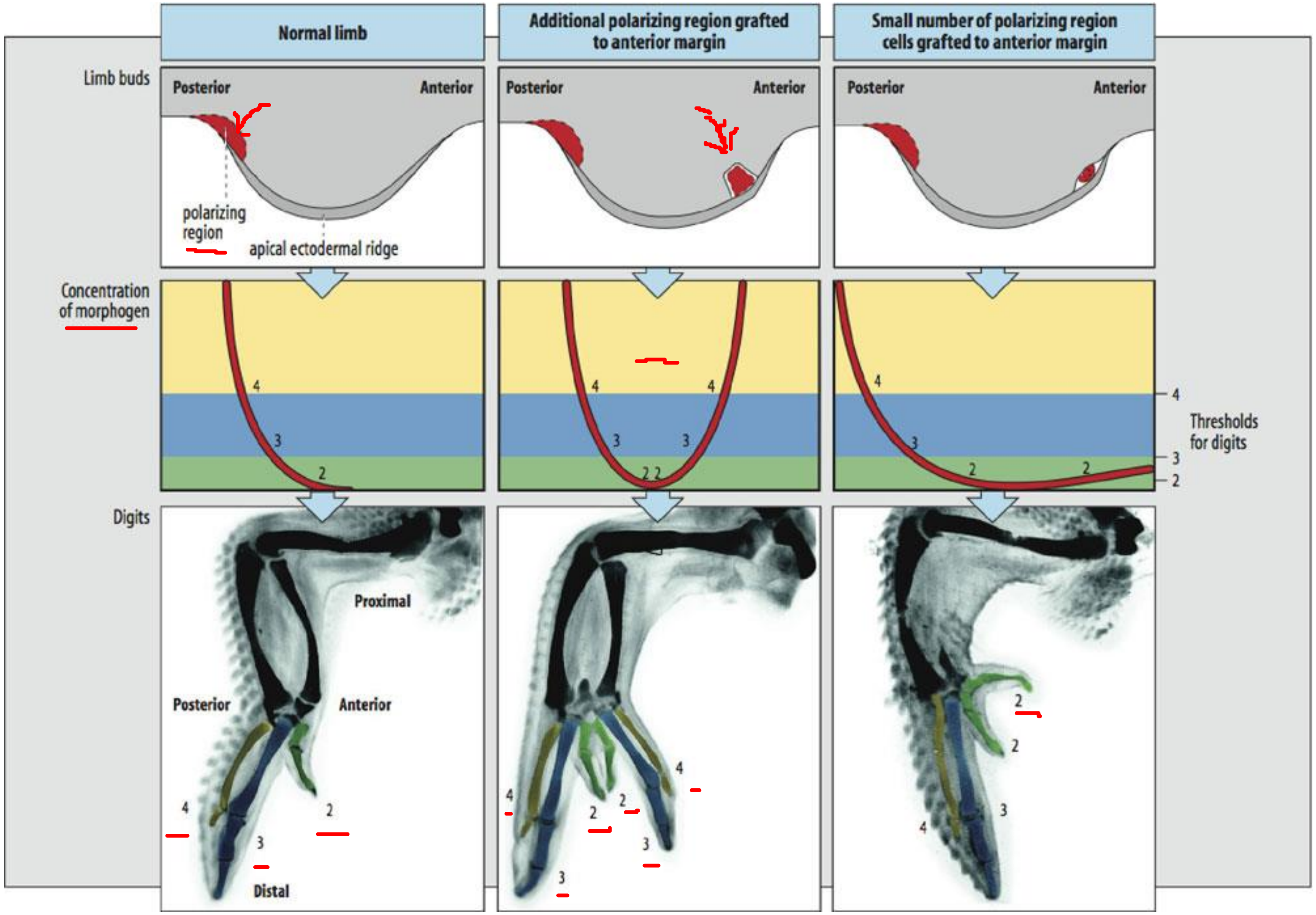




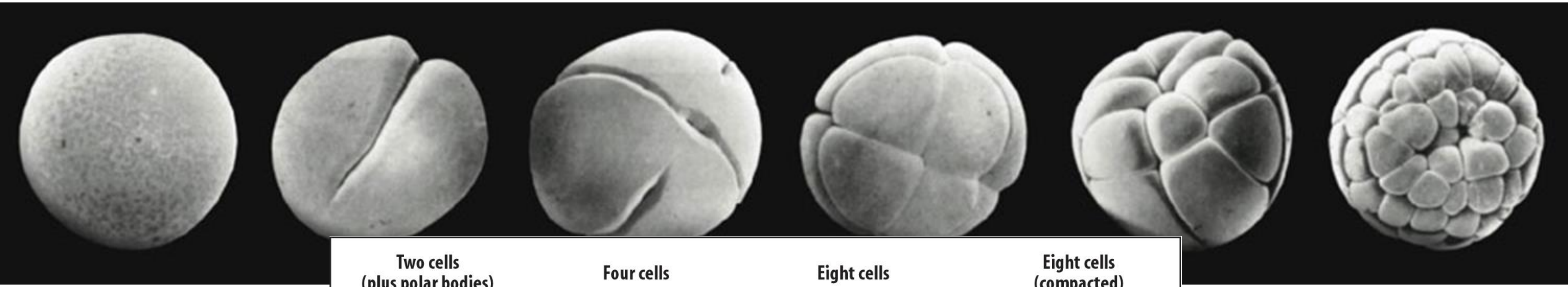


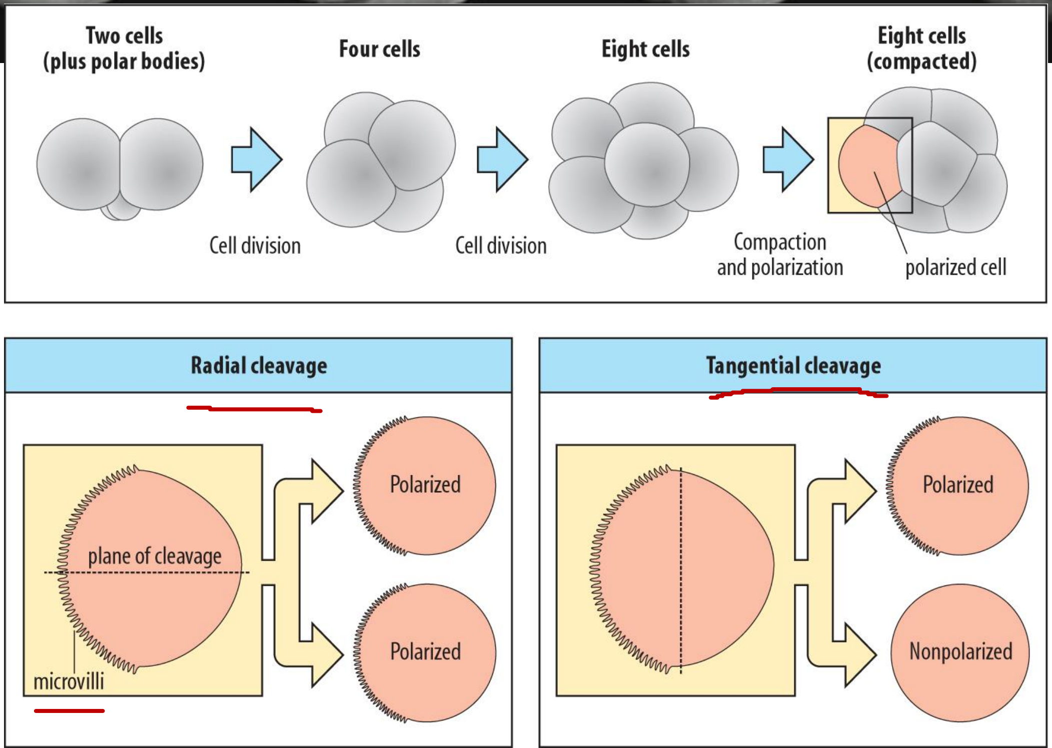
https://www.youtube.com/watch?v=MR79V9UmM6s

Morphogens

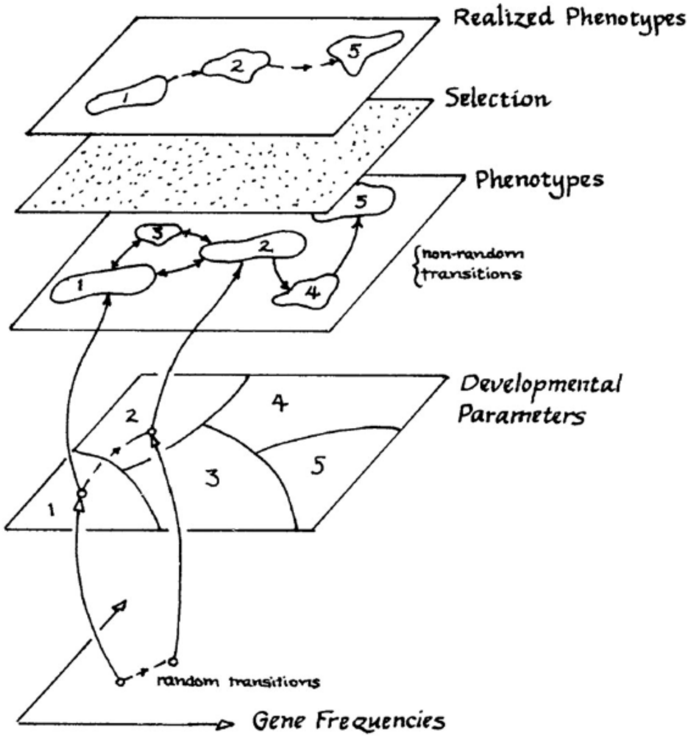

Normal limb
Additional polarizing region grafted to anterior margin
Small number of polarizing region cells grafted to anterior margin
Limb buds
Posterior
Anterior
polarizing region
apical ectodermal ridge
Concentration of morphogen
4
3
2
Thresholds for digits
Digits
Proximal
Distal

Embryogenesis



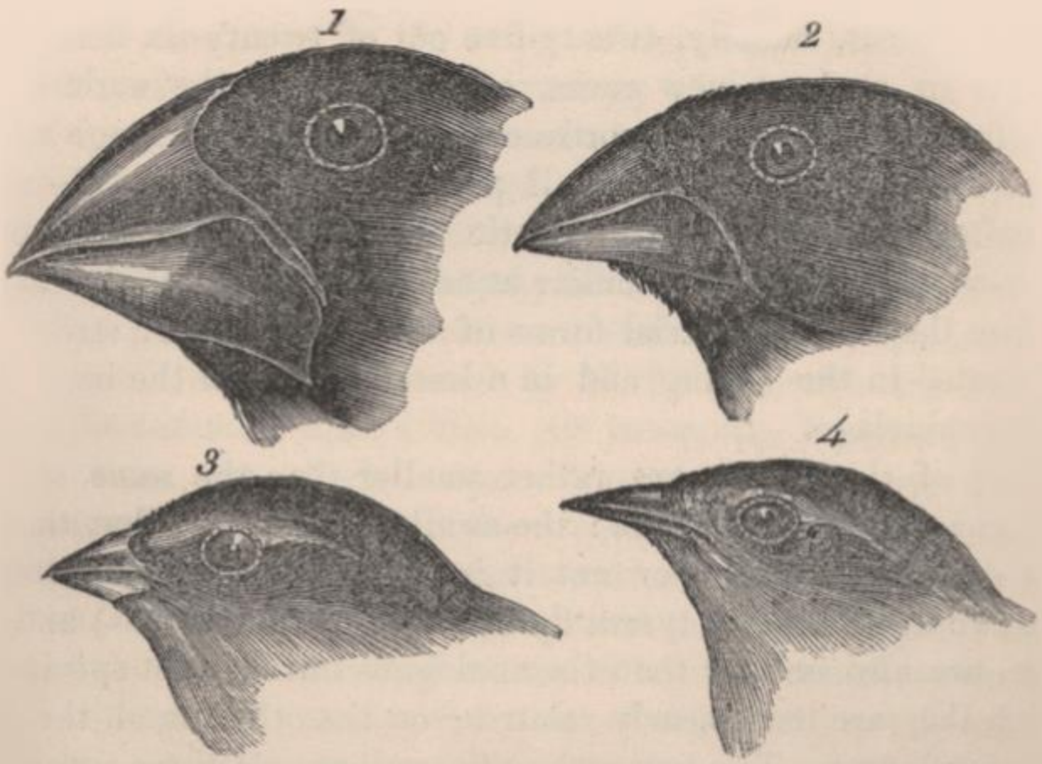


Heterochrony



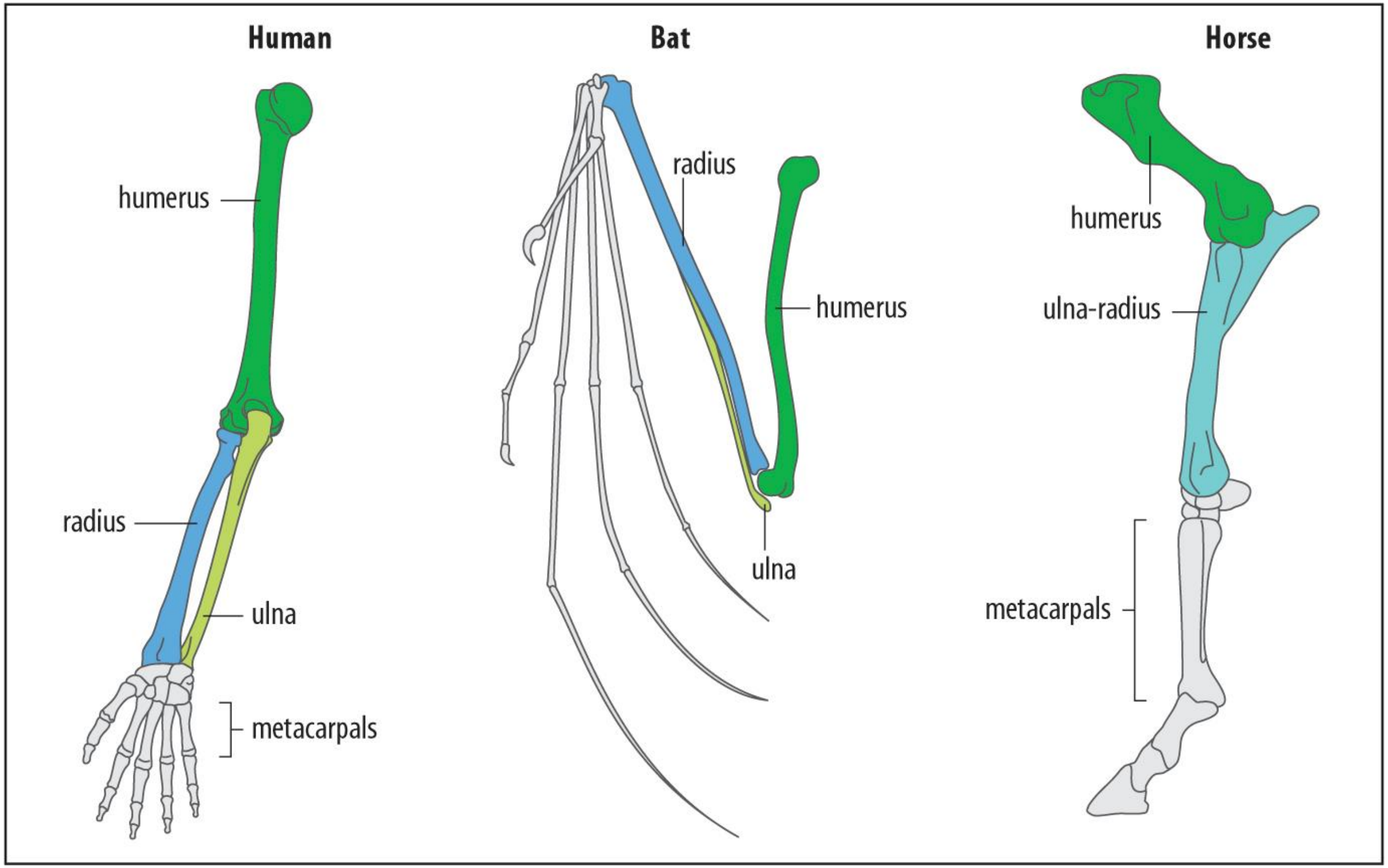
Heterochrony:
Developmental change in timing or rate of events

Evolution of Heterochrony



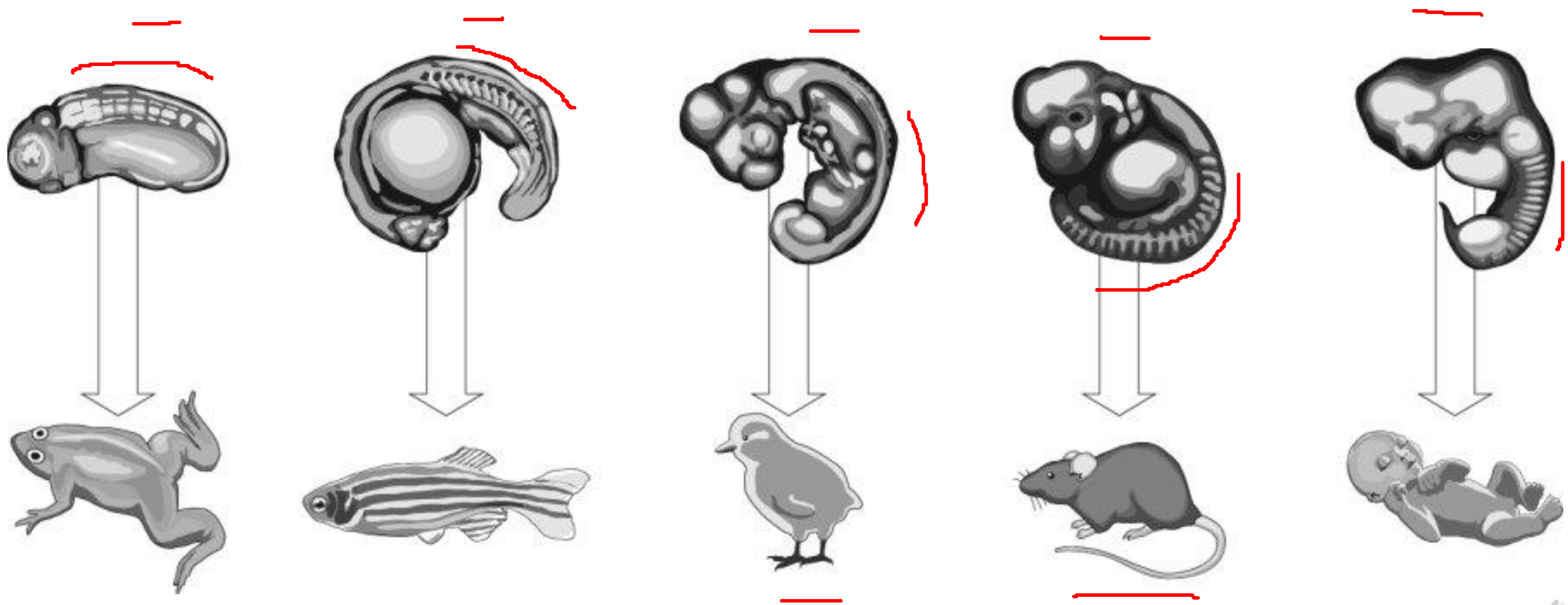
1. Geospiza magnirostris.
3. Geospiza parvula.
2. Geospiza fortis.
4. Certhidea olivacea.

Homologous Evolution

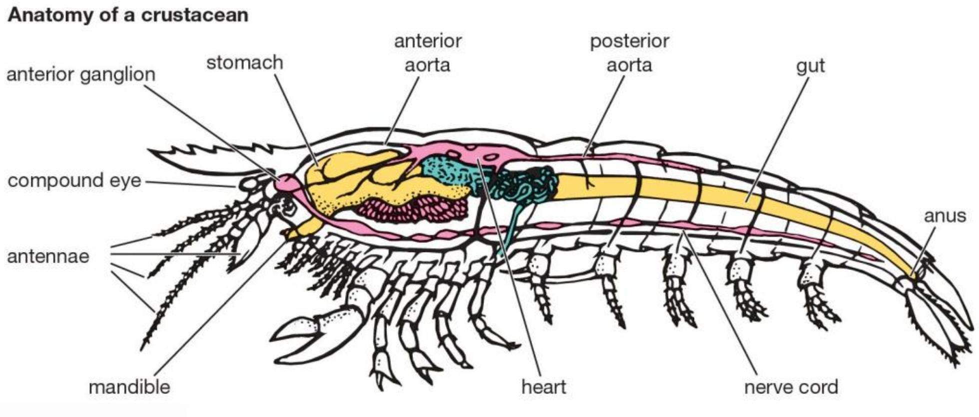


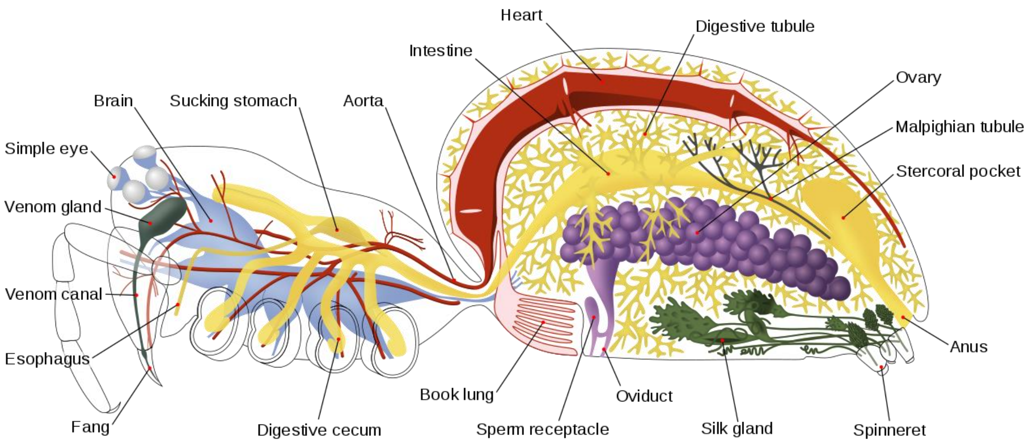
Conserved Developmental plans

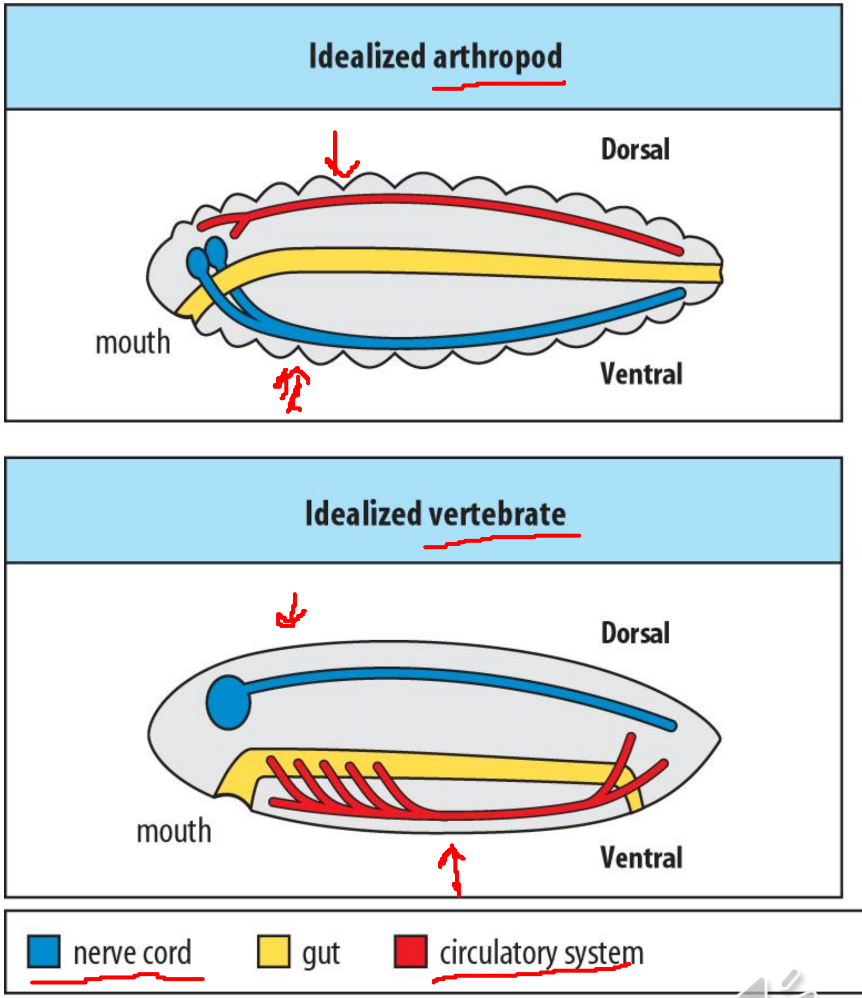
- Phylotypic Stage



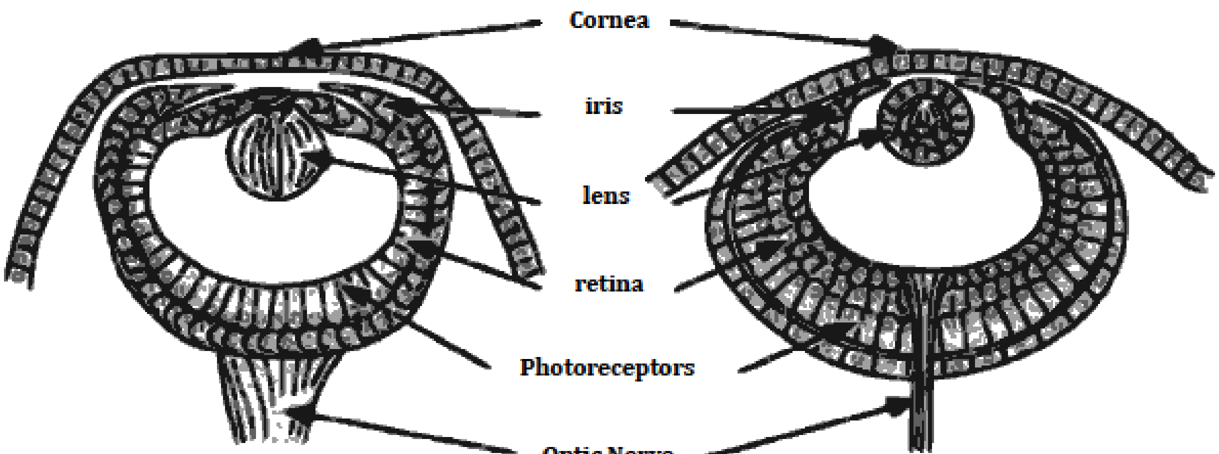
Body organization





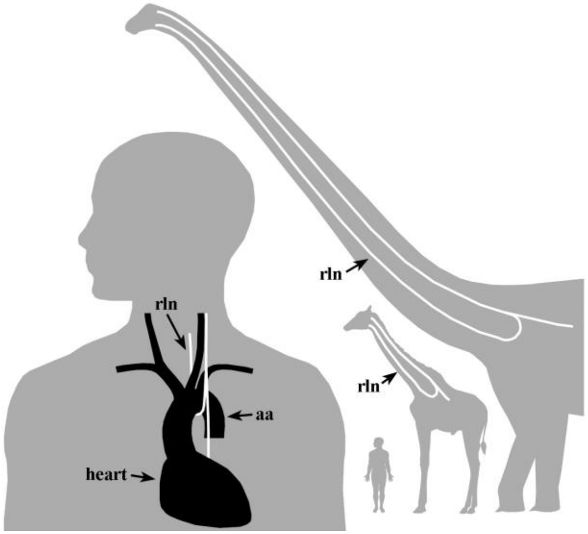


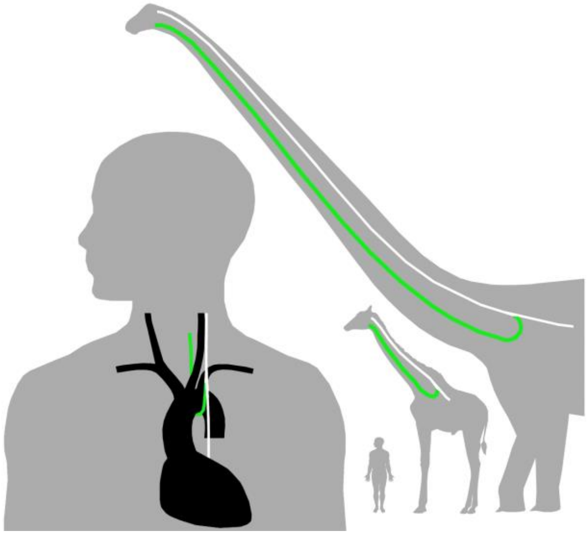
Locked in Imperfection (1)



Cephalopods Mammals

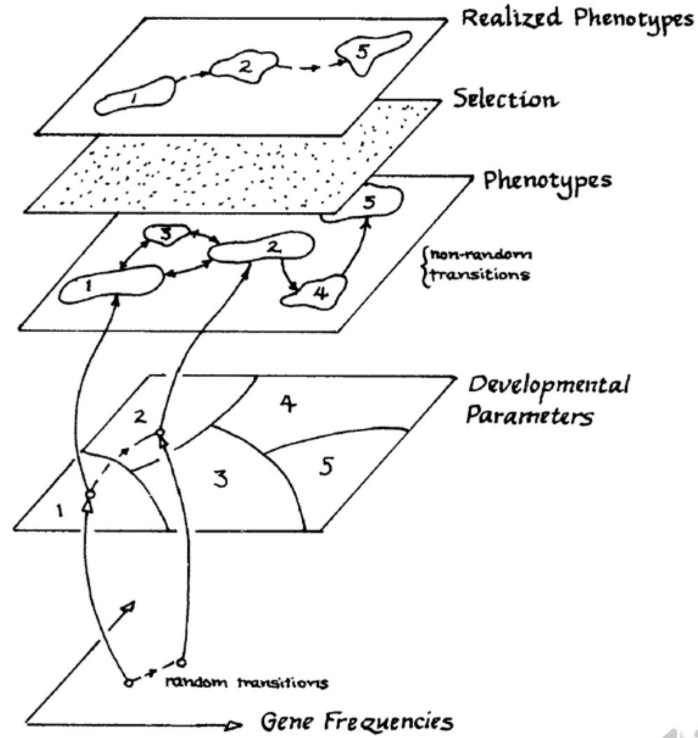
Locked in Imperfection (2)





Summary concepts on development

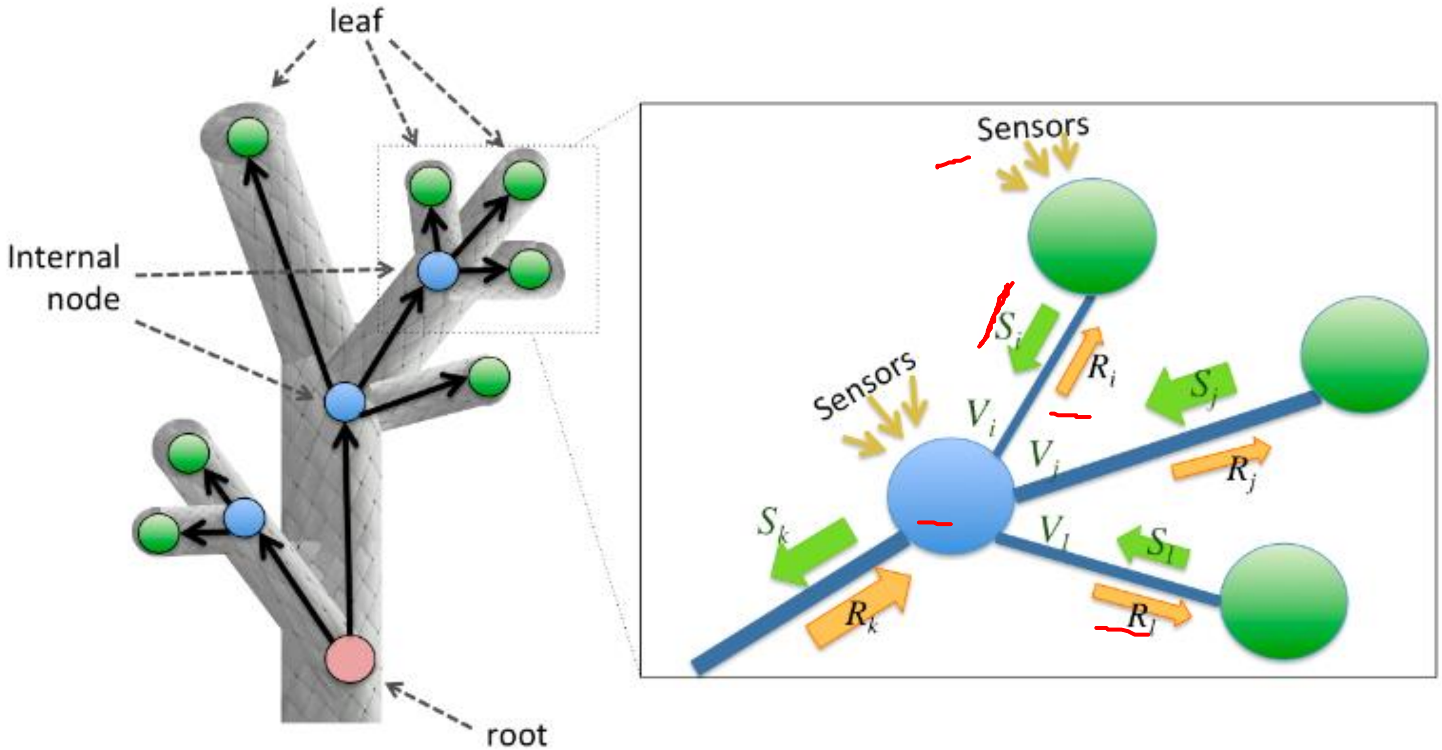
- Morphogens: chemicals that influence development
- Heterochrony: timed expression of morphogens
- Homologous evolution: Shared genetic ancestry



Chemistry inspired developmental systems

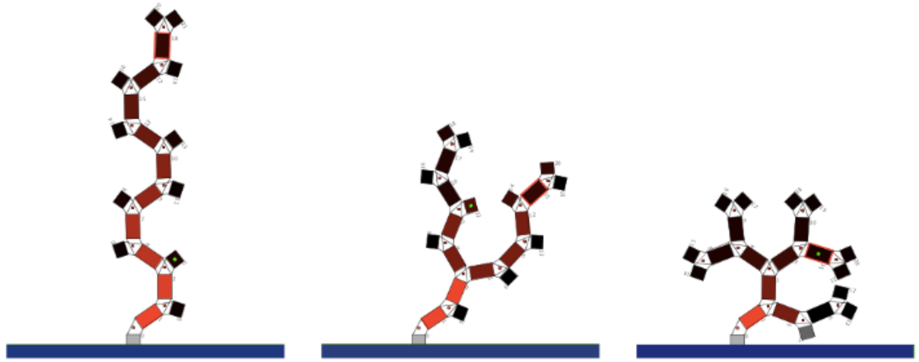
- Morphogens
- Reaction-diffusion: Exercise using *Gray-Scott algorithm*

Vascular Morphogenesis



Zahadat et al., 2017, Vascular Morphogenesis Controller: A Generative Model For Developing Morphology of Artificial Structures

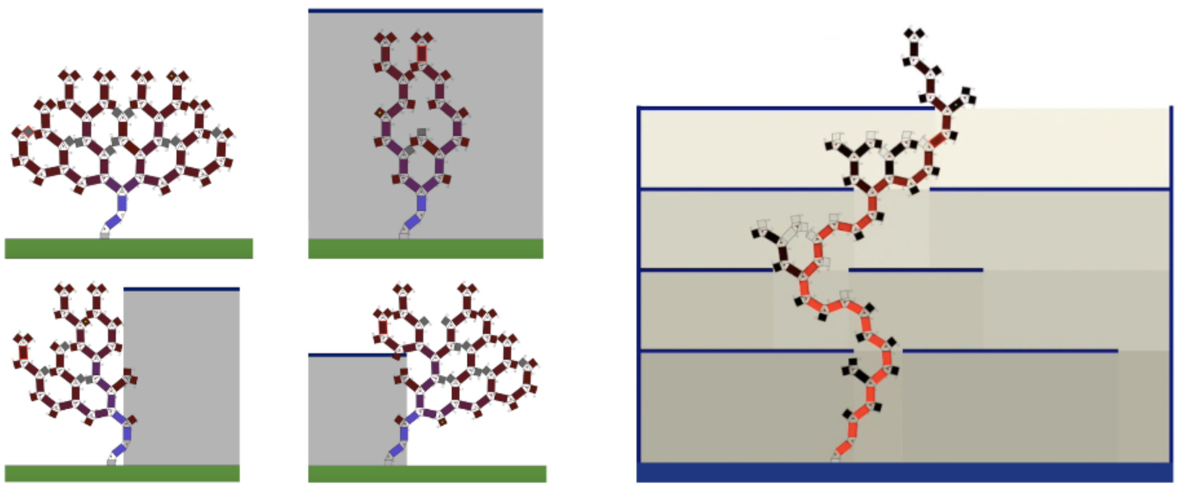
Morphogenesis through light sensing



(a) growing tall in a calm environment

(b) growing tall in a harsh environment

(c) growing bushy



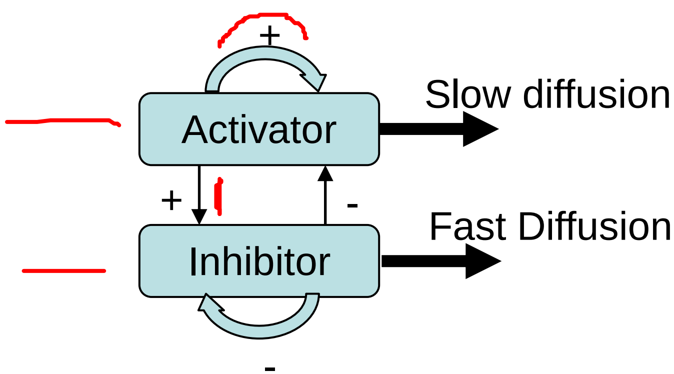
(d) identical parameter set grows the structure differently in different light setups

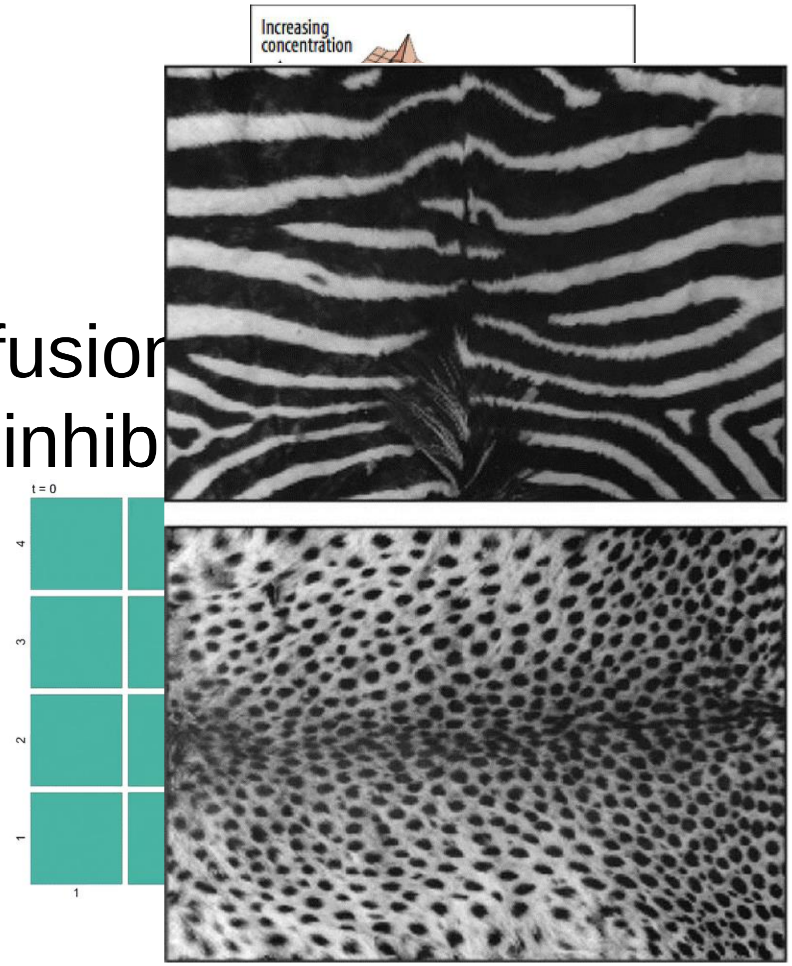
(e) finding the brightest layer in a layered environment

Zahadat et al., 2017, Vascular Morphogenesis Controller: A Generative Model ForDeveloping Morphology of Artificial Structures

Morphogens

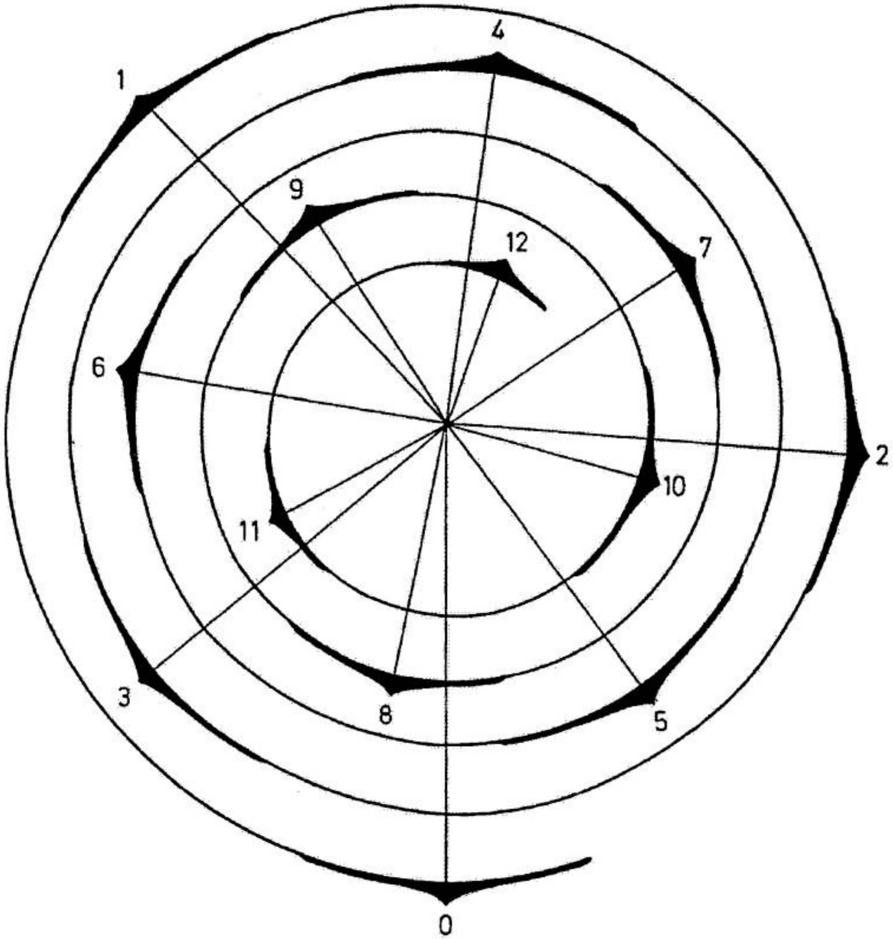
- Reaction-diffusion through self-inhib

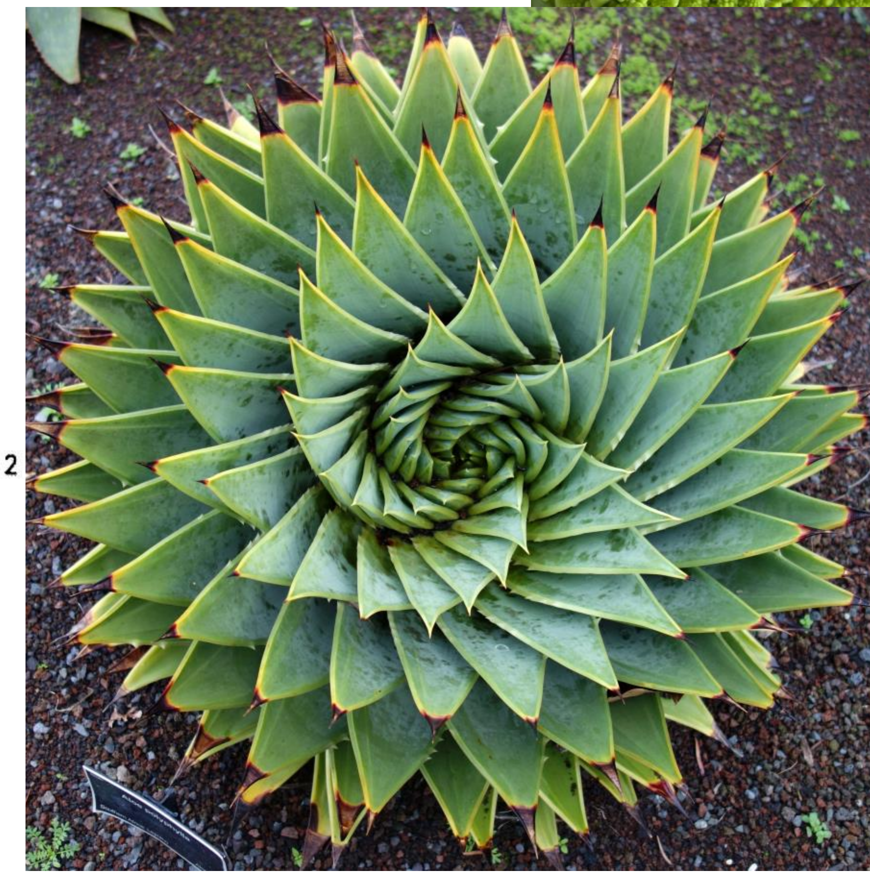




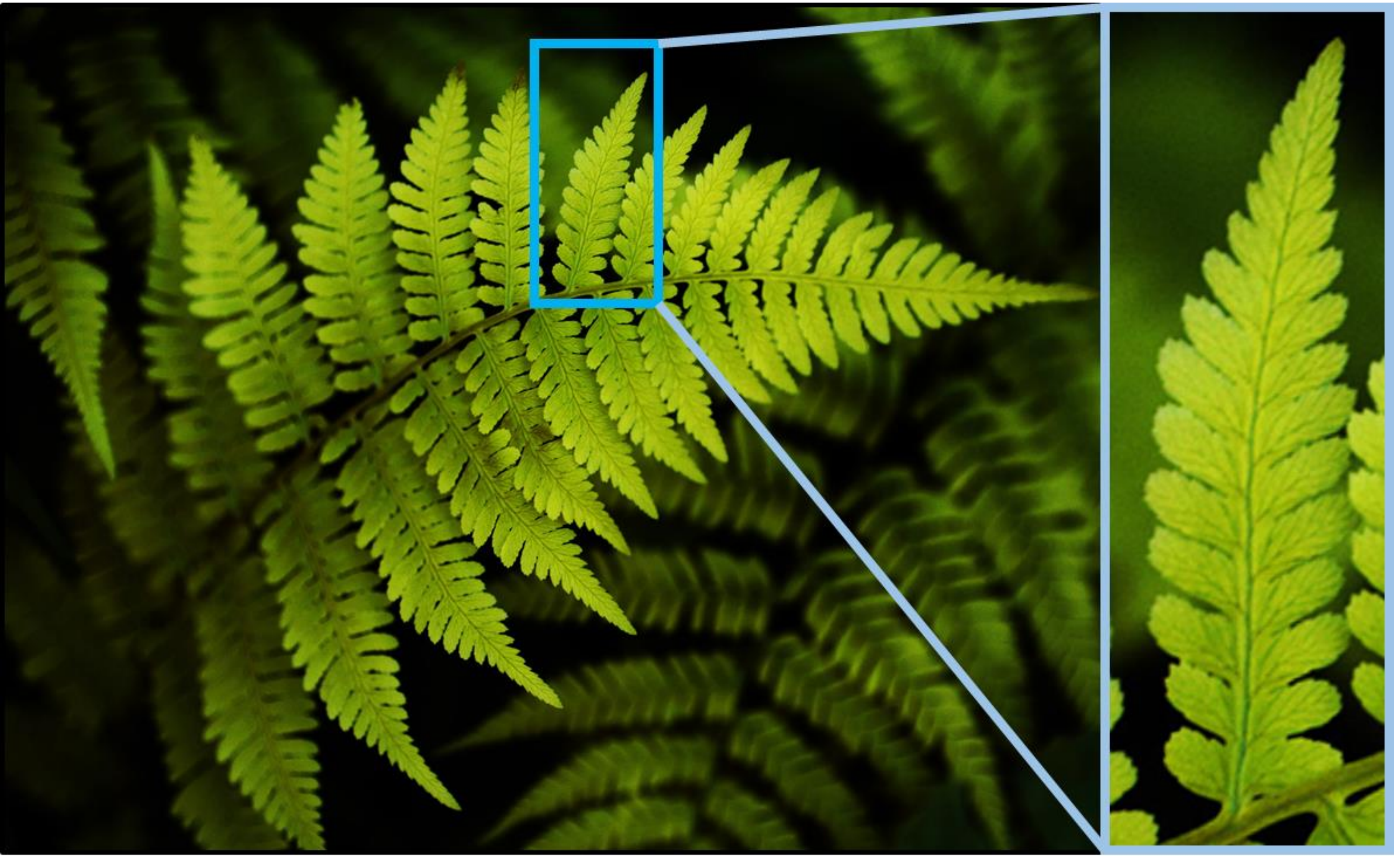
https://www.youtube.com/watch?v=MR79V9UmM6s

Leaf Phyllotaxis

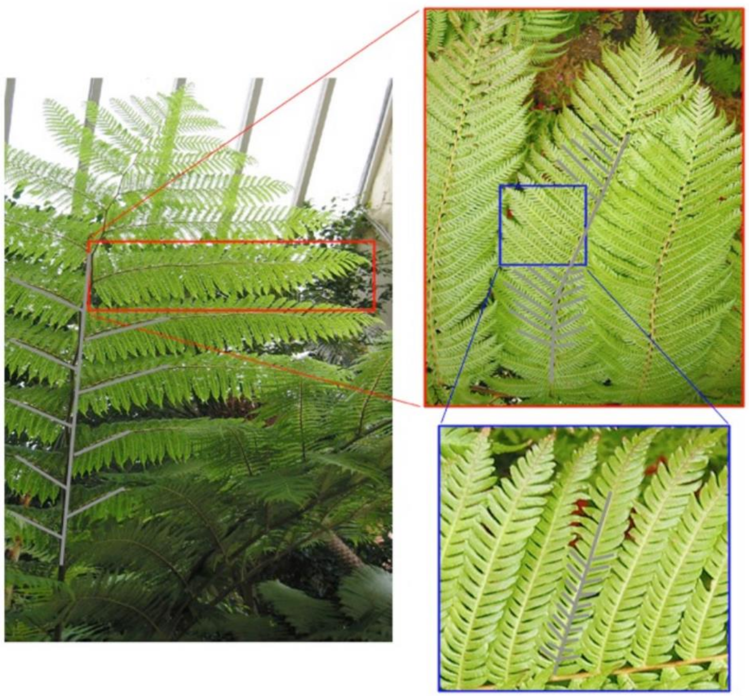


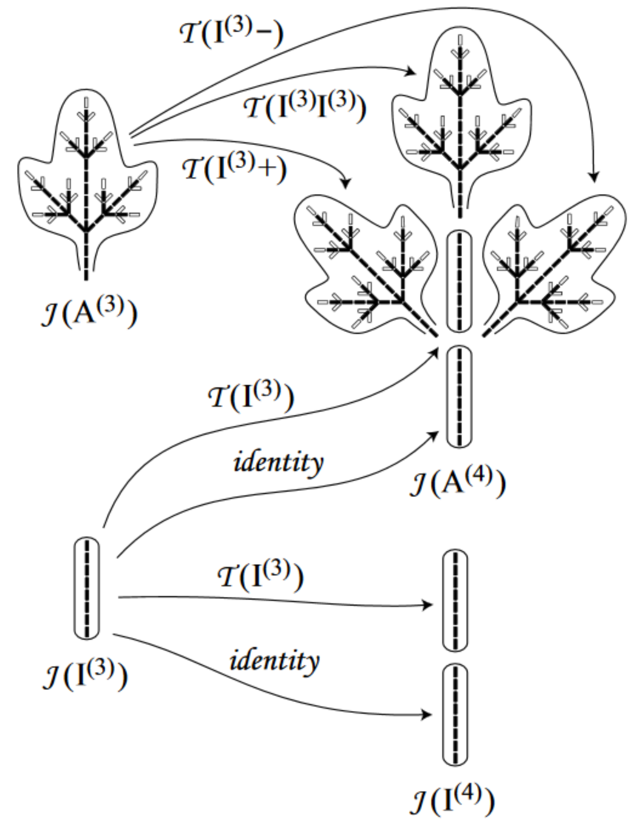


Repetition, Self-Similarity and Modularity



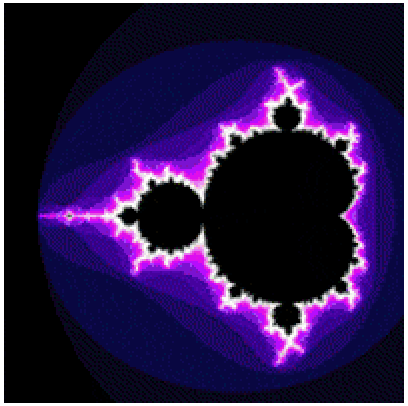
Repetition, Self-similarity and Modularity





Przemyslaw Prusinkiewicz, 2014, Self−Similarity In Plants: Integrating Mathematical
And Biological Perspectives

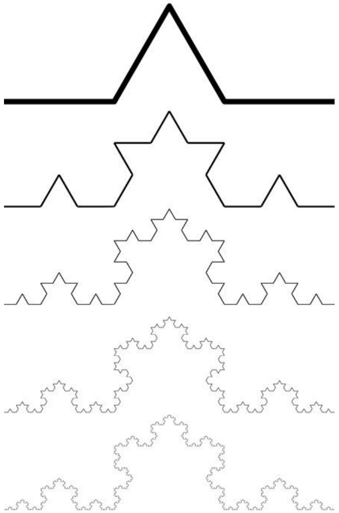
Fractals



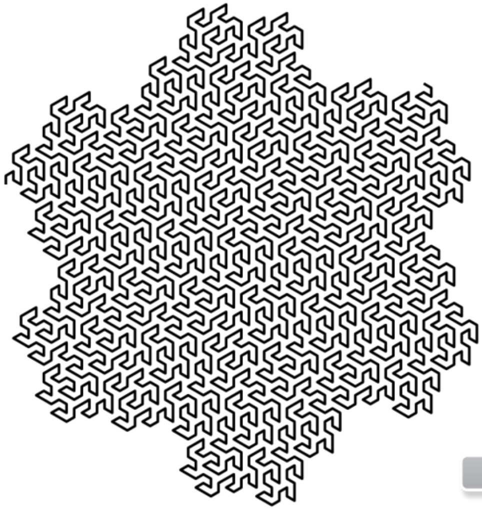
- Mandelbrot set
  - $f(c)z = z^2 + c$
    - Behavior of 0 under iteration f(c)
    - c is a complex number

Rewriting systems:

- Koch-curve



- Flow-snake
  - (Gosper-curve)



Artificial Development

- Rewriting systems
  - L-System
- Chemistry based approaches
  - Turing patterns

- Rewriting Systems
  - Sequential (Formal Grammar)
  - Parallel (Lindenmayer-System)

$$A \rightarrow AB$$

$$B \rightarrow A$$

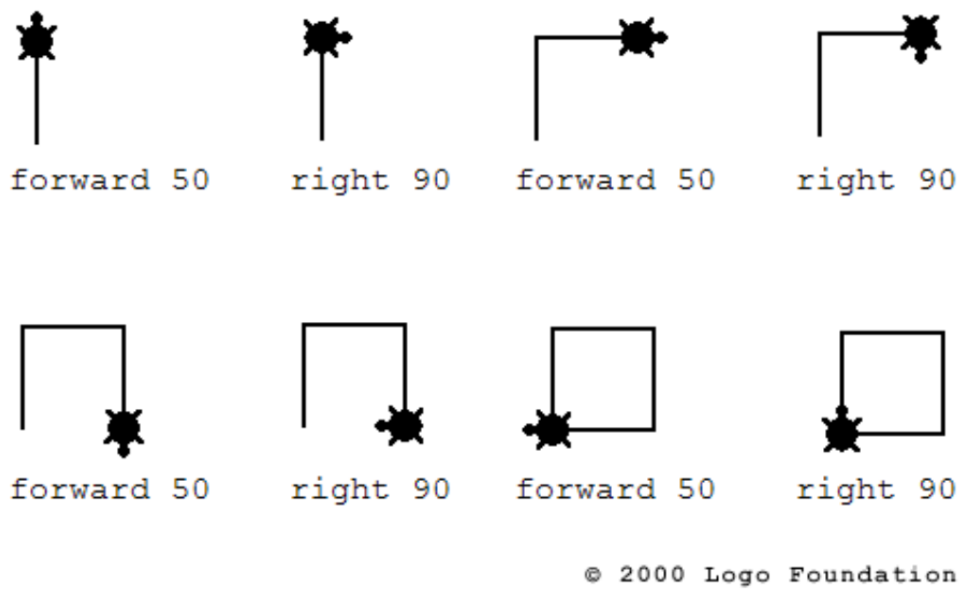
Simple rewrite	**L-system**
AB	AB
AA	ABA
ABAB	ABAAB
AAAA	ABAABABA
ABABABAB	ABAABABAABAAB

An example L-System

- Alphabet: {a,b}
- Production rules (grammar)
  - a->ab
  - b->a
- Axiom b
- Each symbol is replaced once each iteration

L-Systems for generating graphics: turtle graphics

- Read the string produced by the L-System from left to right changing the state of the turtle



Example L-system for Drawing

- Alphabet: {F, f, +, -}
- F: move the turtle forward (drawing a line)
- f: move the turtle forward (don't draw)
- +/-: turn right/left (by some angle)

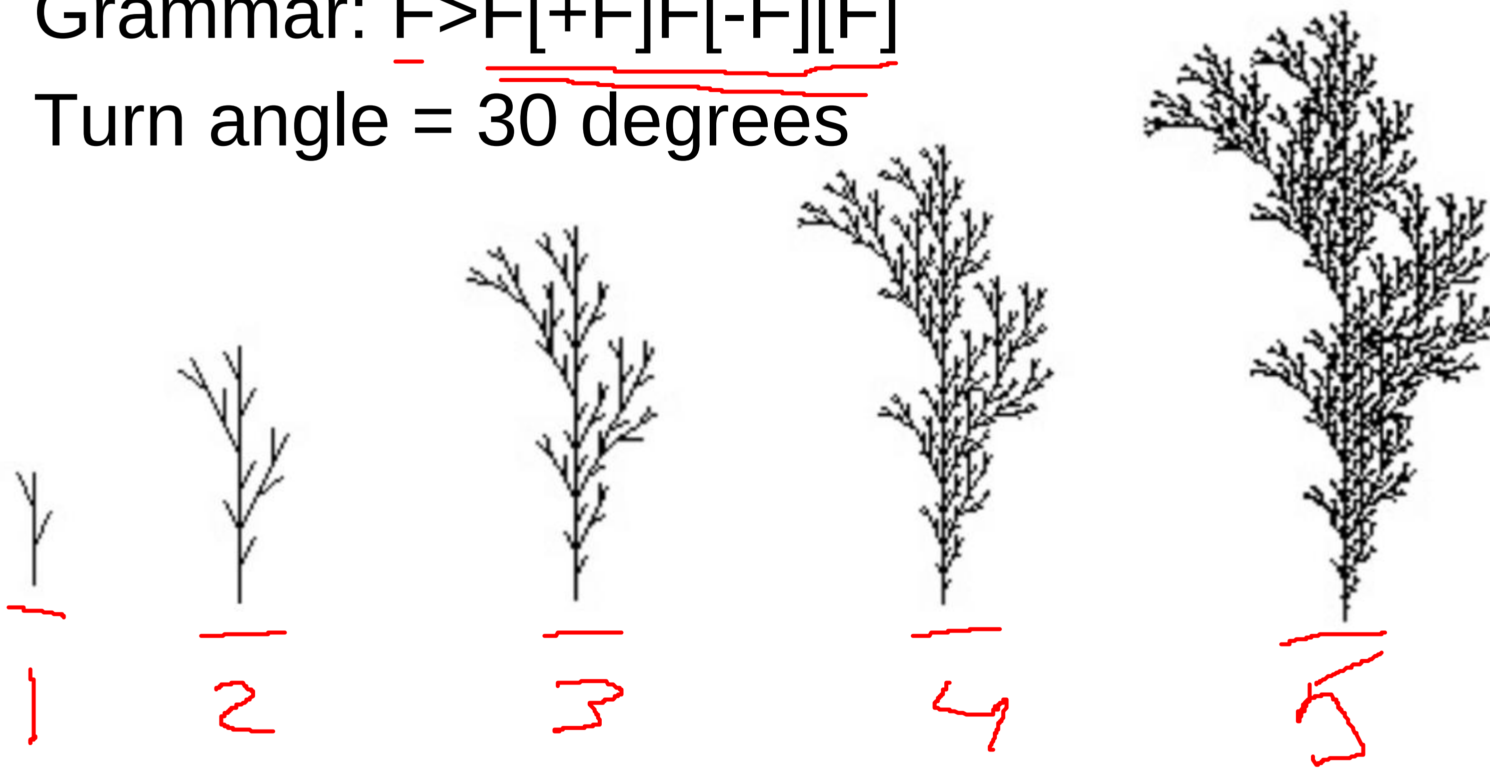
- What would FfFfFfFf do?
- What would F+F+F+F+ do?

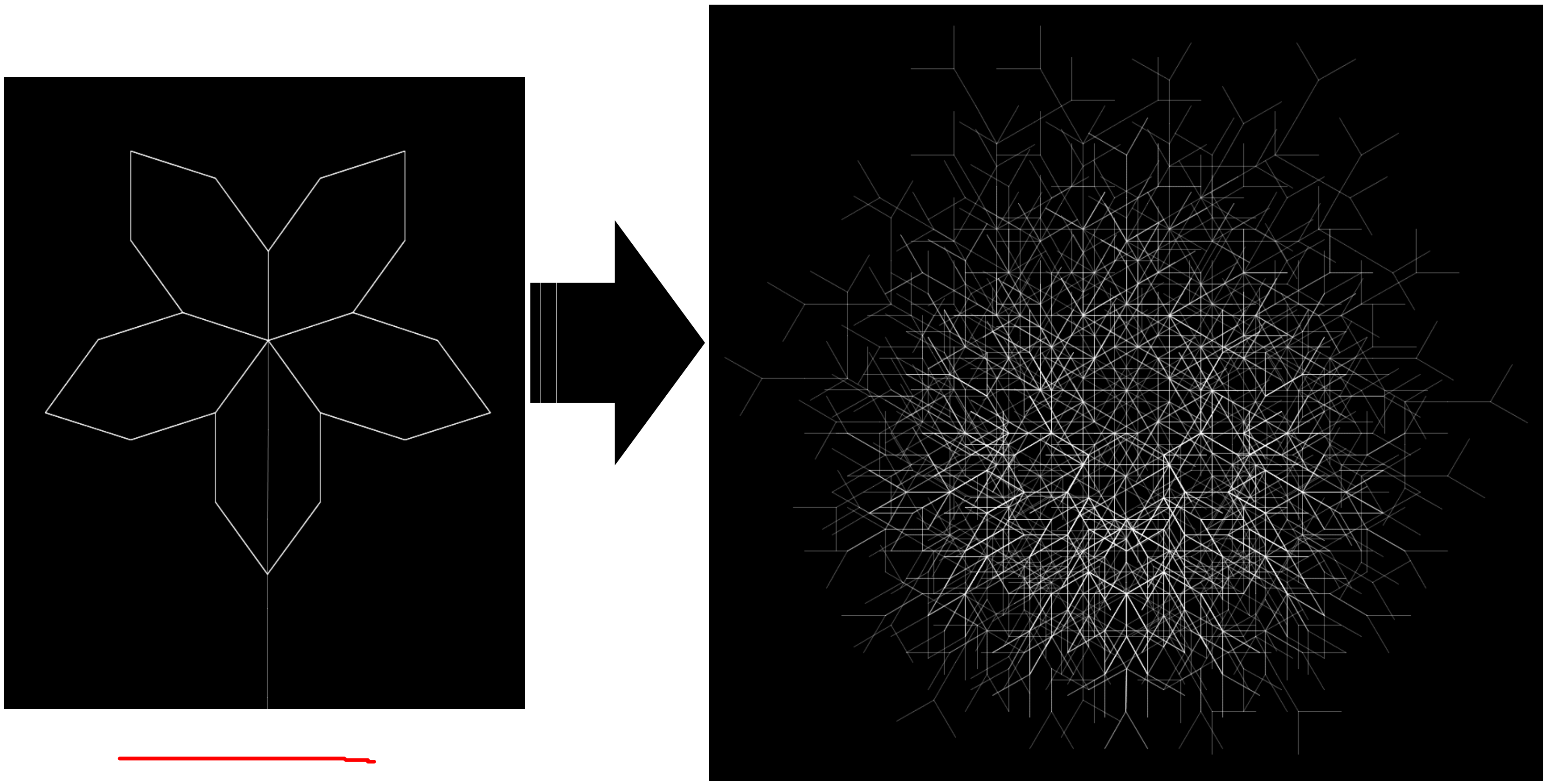
Bracketed L-Systems

- Alphabet: {F, f, +, -, [,]}
- [: push the current state (x, y, heading of the turtle) onto a pushdown stack
-]: pop the current state of the turtle and move the turtle there without drawing
- Enables branching structures!

L-System

- Axiom: F
- Grammar: F>F[+F]F[-F][F]
- Turn angle = 30 degrees





Come up with your own rules!

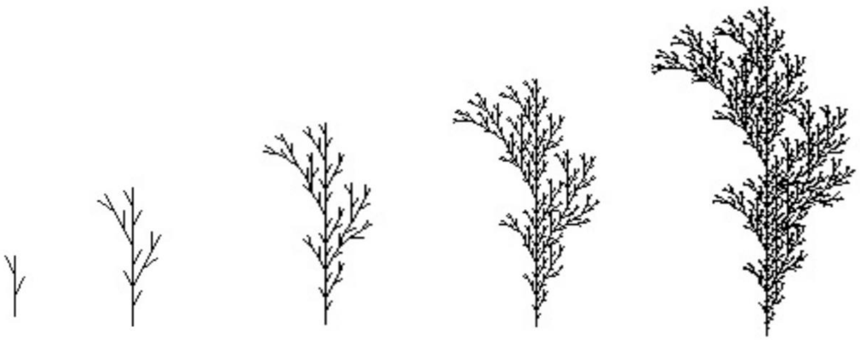
Go to: **https://onlinemathtools.com/l-system-generator**

- Alphabet: {F, f, +, -, [,]}
- [: push the current state (x, y, heading of the turtle) onto a pushdown stack
-]: pop the current state of the turtle and move the turtle there without drawing

Axiom: F

Grammar: F>F[+F]F[-F][F]

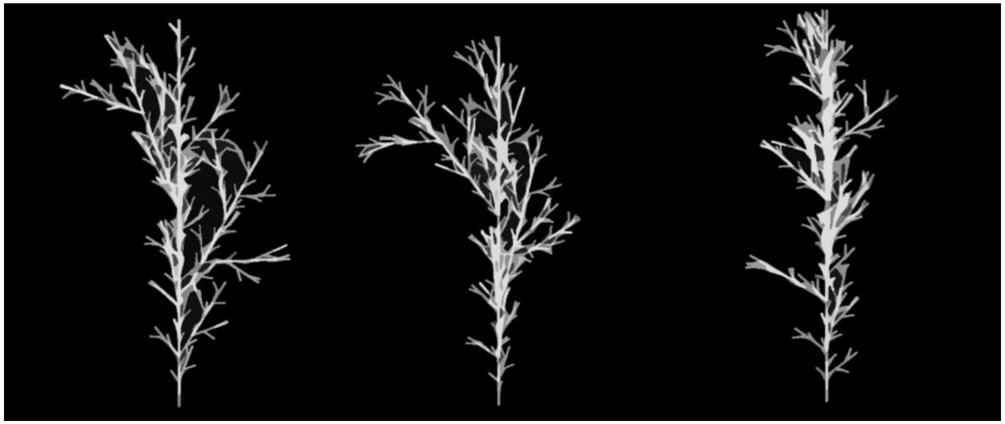
Turn angle = 30 degrees

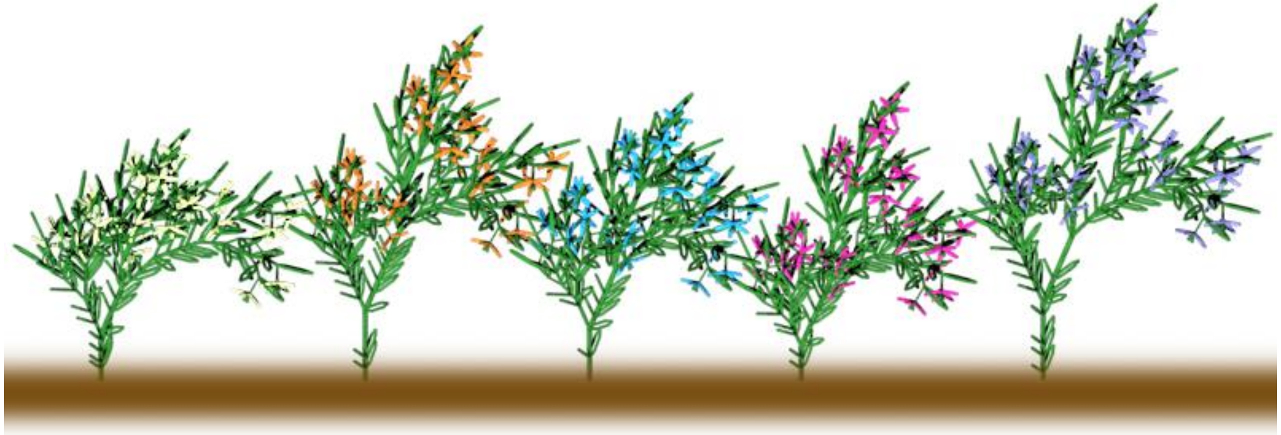


Types of L-Systems

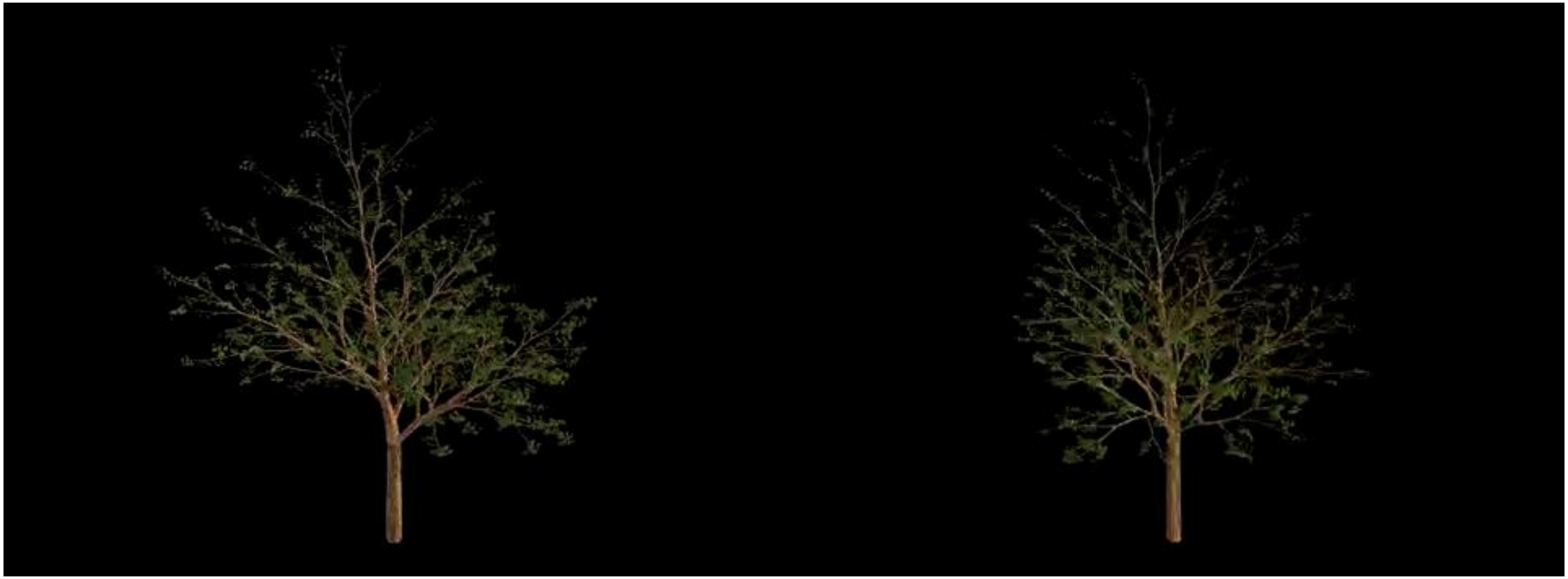
- Context free: production rules refer only to an individual symbol
- Context-sensitive: production rules can depend on the symbol’s neighbors
- Deterministic: one production rules for each symbol
- Non-deterministic: several production rules for a symbol
- Parametric: a symbol references a parameter list

Examples of Stochasticity



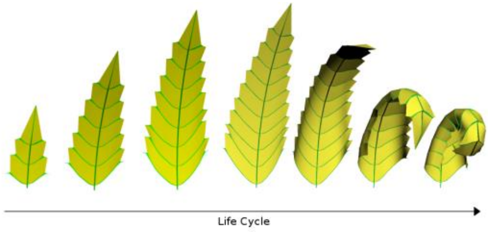


Ecomod by Aleš Zamuda



https://www.youtube.com/watch?v=O0HUQR1mTbg
https://www.youtube.com/watch?v=qiogqd2PlW0

Unity's Speed Tree



Peyrat *et al.* 2008

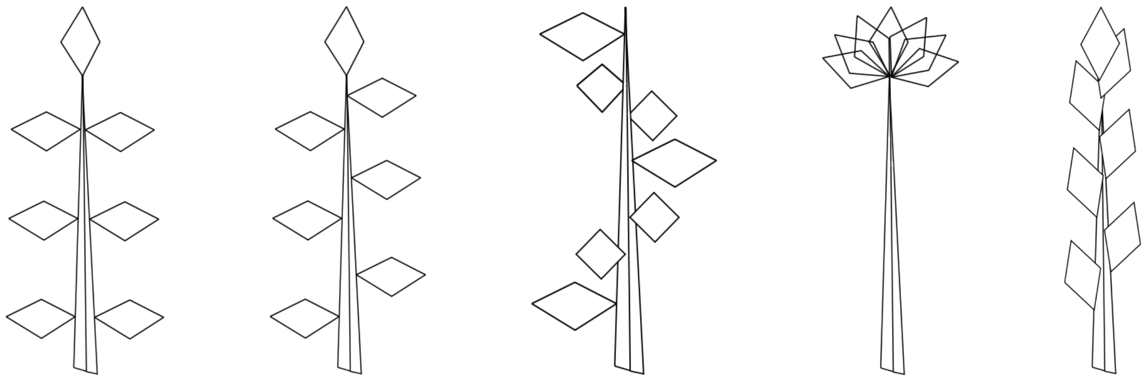


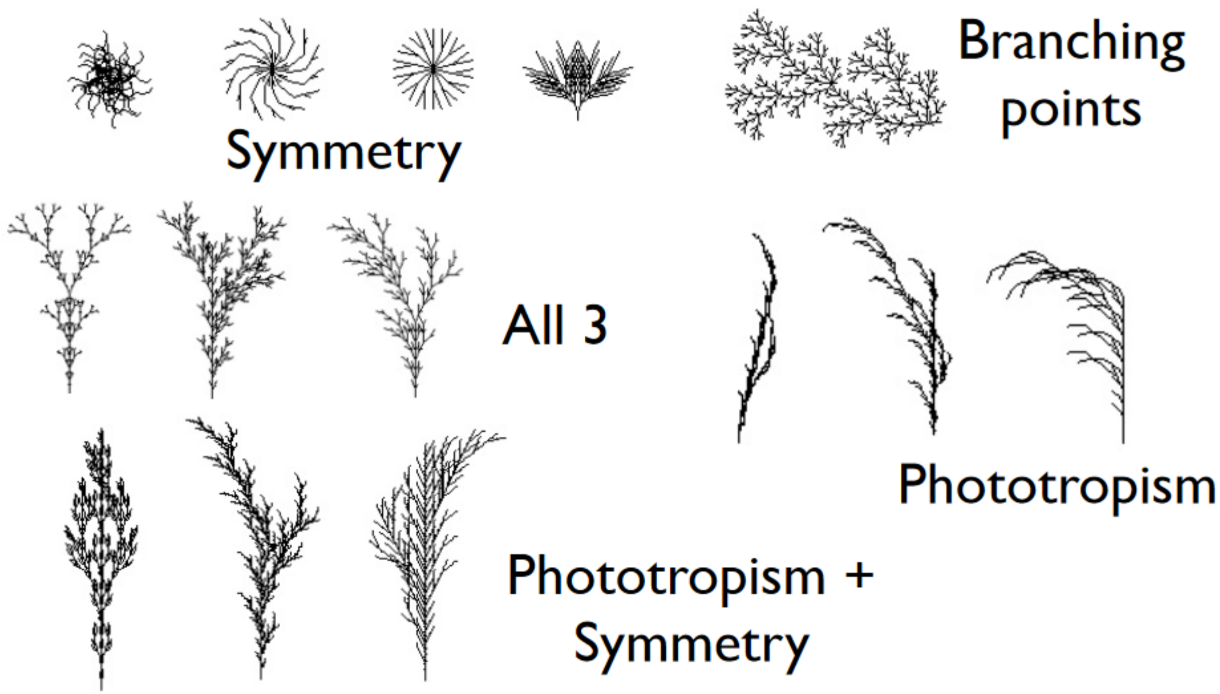
Figure 7: Distributions of leaves, from left to right: stacked (1), staggered (2), spiral (3), bunched (4), and coniferous (5).

Image taken from: Aleš Zamuda and Janez Brest (2014) Vectorized Procedural Models for Animated Trees Reconstruction using Differential Evolution

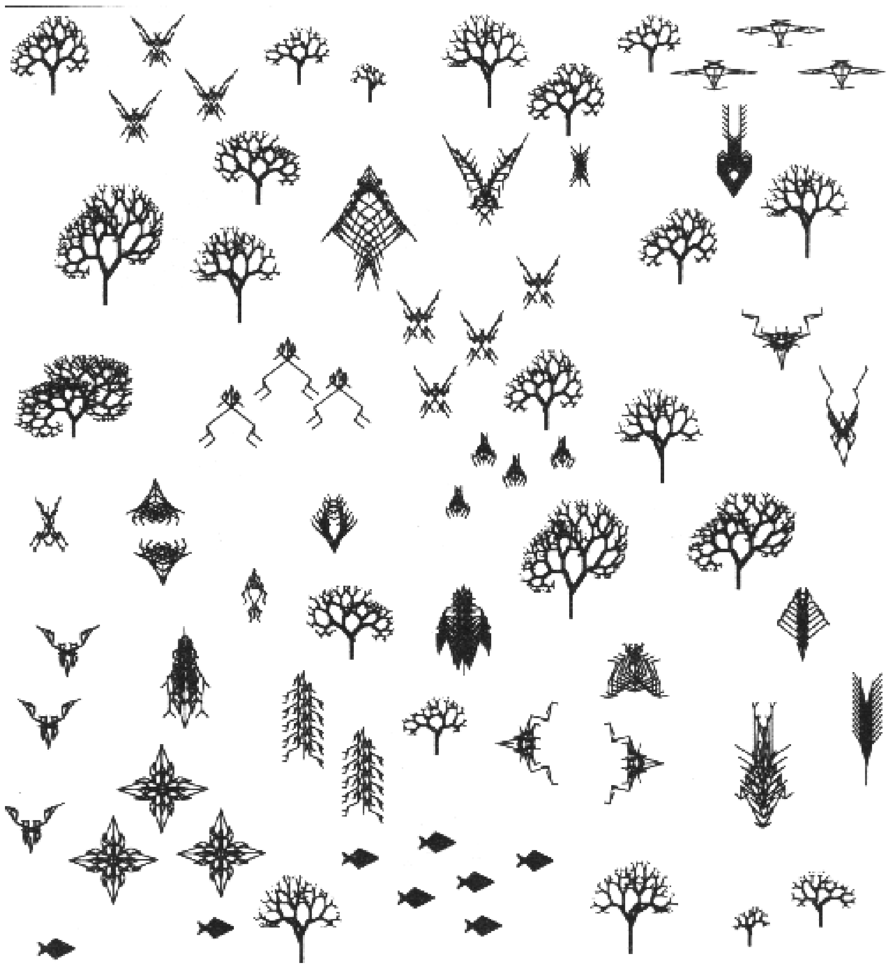
Evolving L-Systems

- Mutation
  - Axiom
  - Production Rules
  - Probabilities
  - Angles
- Fitness
  - Phototropism
  - Bilateral symmetry
  - Proportion of branching points

Evolved L-Systems



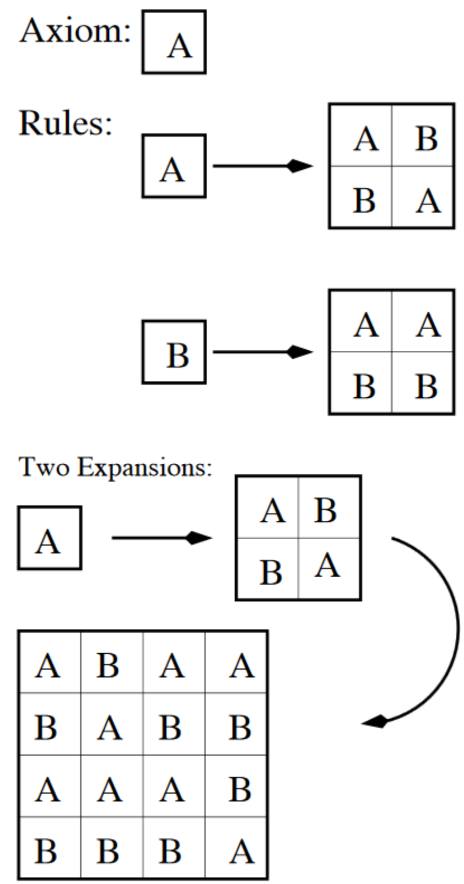
Evolved L-Systems



https://www.codeproject.com/Articles/17387/AI-Dawkins-Biomorphs-And-Other-Evolving-Creatures

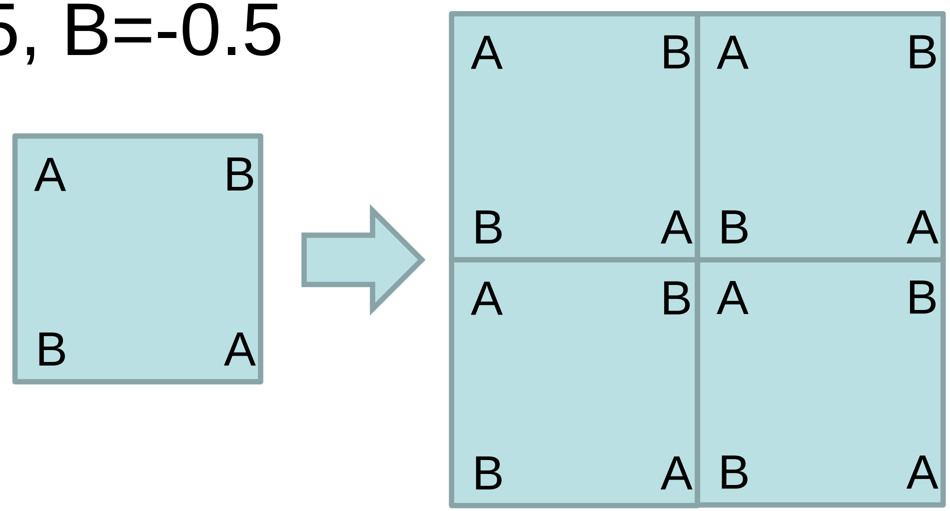
2D L-Systems

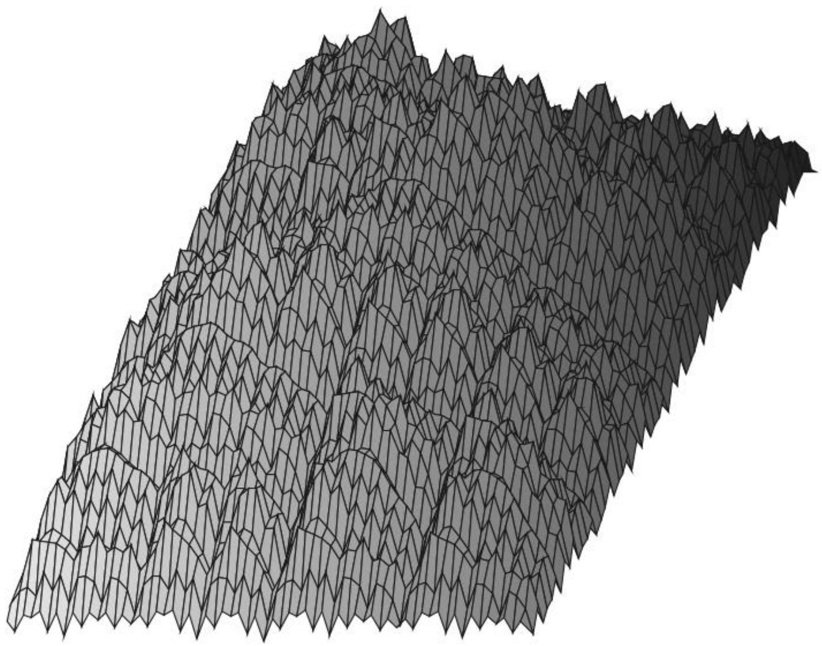
- (Matrix rewriting systems)



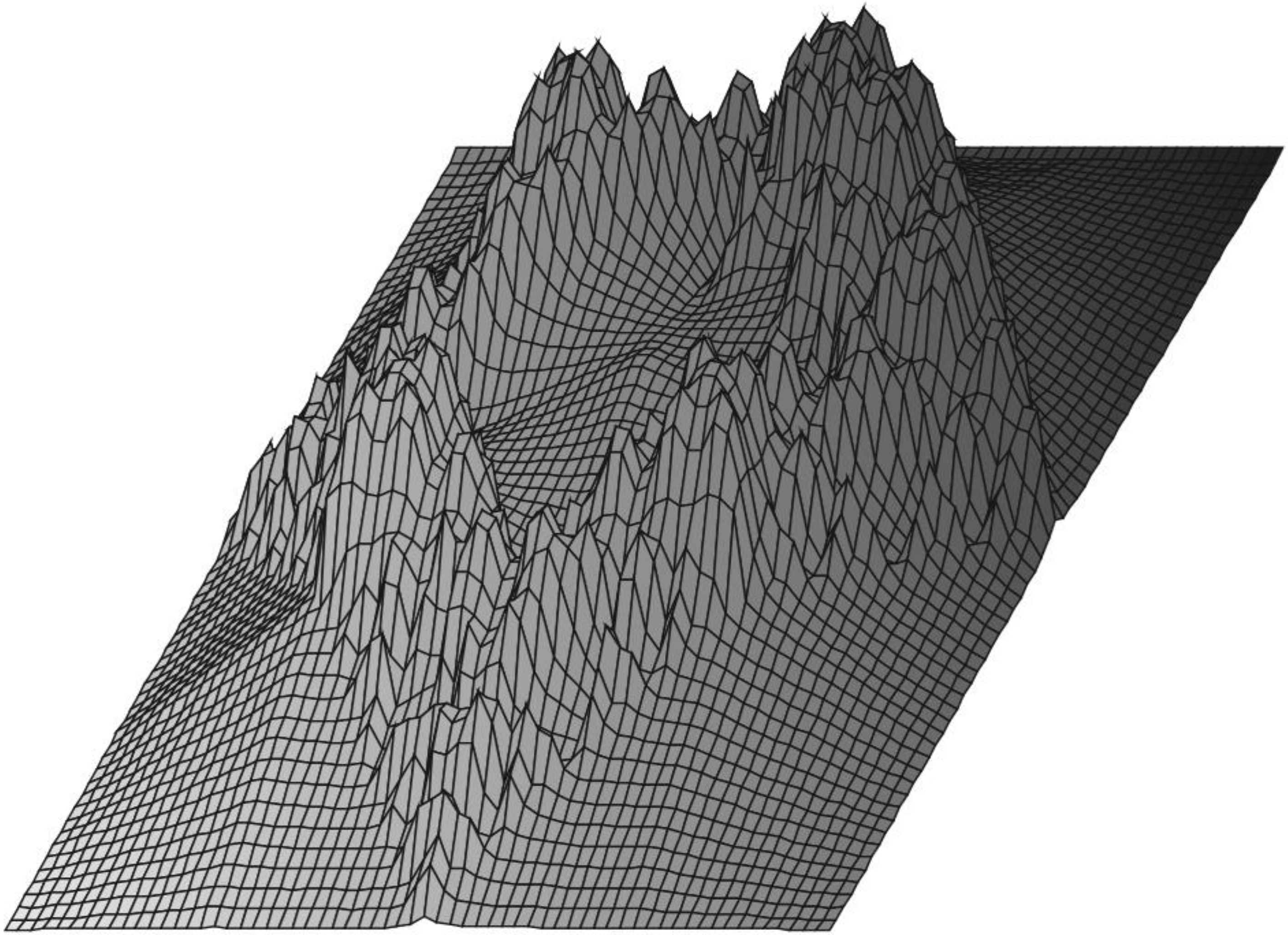
Terrain interpretation of 2D L-Systems

- Letters are interpreted for lowering or raising corners of a square
- E.g. A=+0.5, B=-0.5

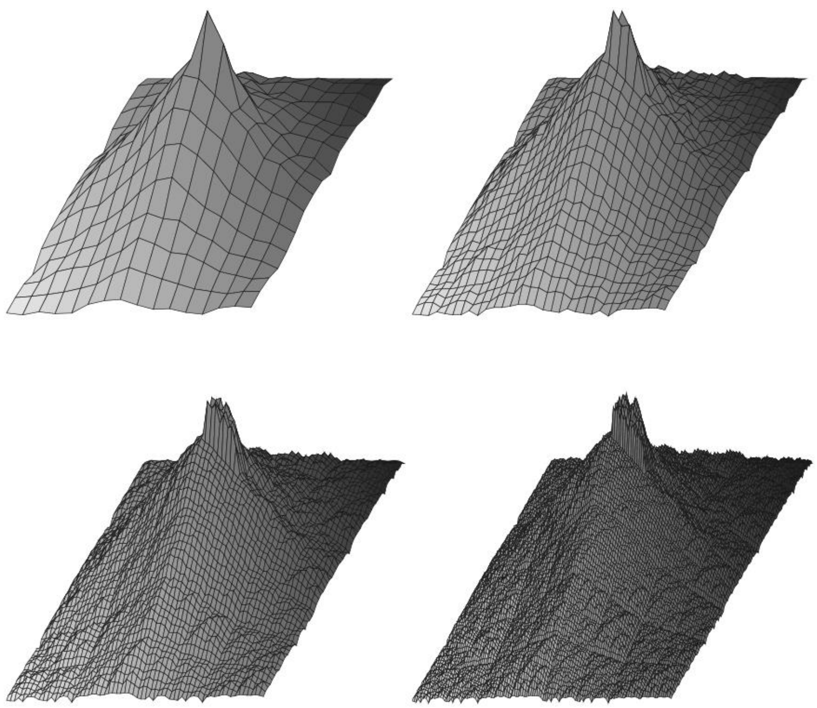




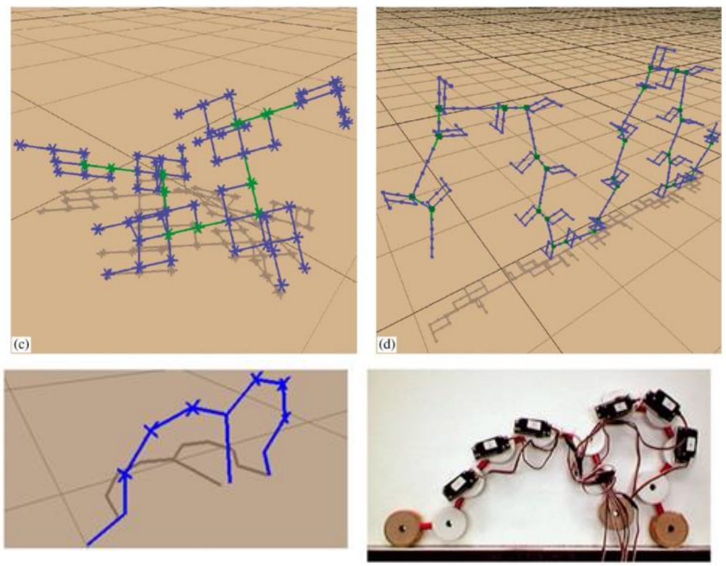
Six rewritings of A>ABBA, B>AABB



Infinite resolution!

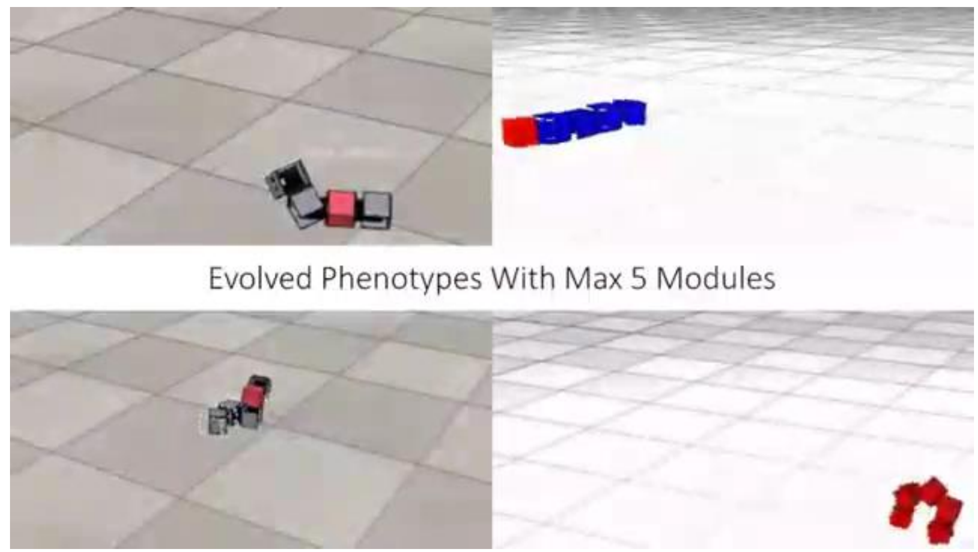


Evolving robots created from L-System



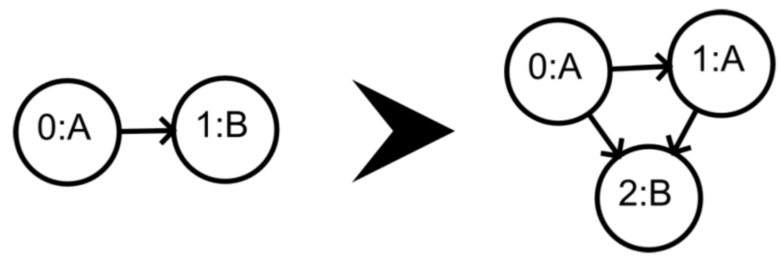
Hornby and Pollack 2001

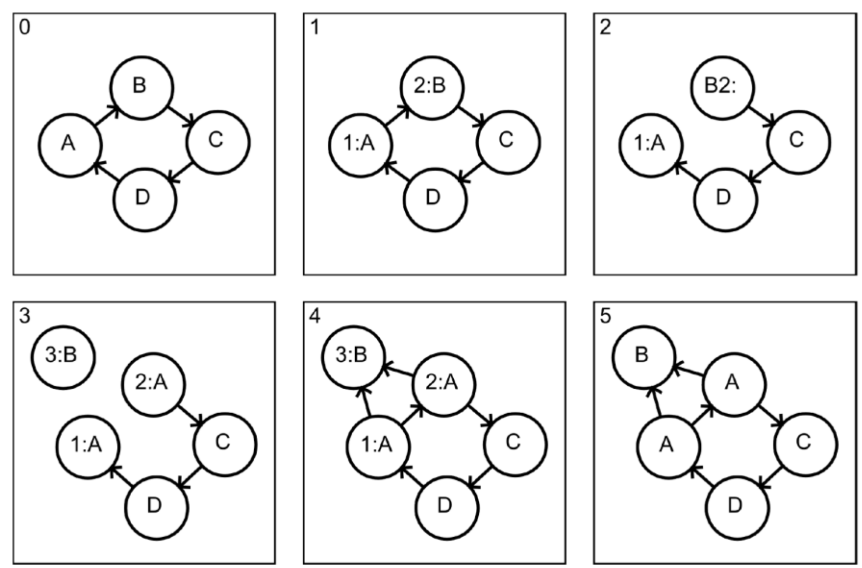
Direct Encoding L-System



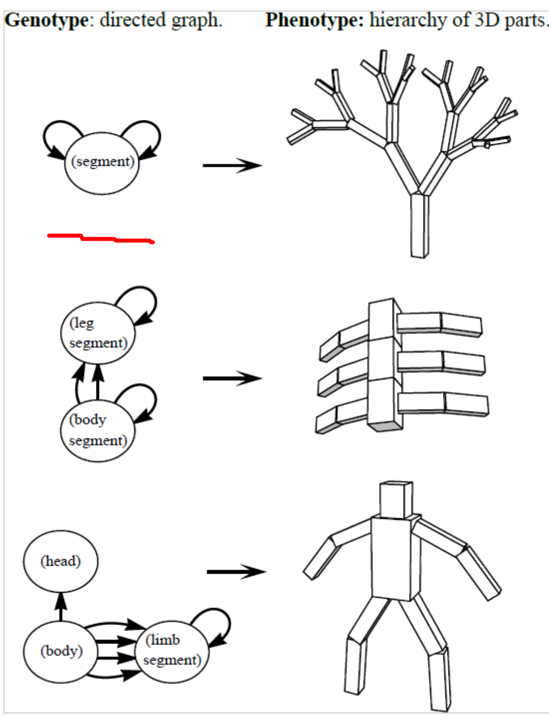
Rewriting Graph Grammars

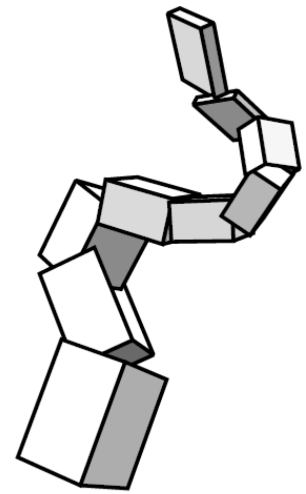
- 1: Identify left-hand subgraph
- 2: Remove edges
- 3: Transform graph
- 4: Copy edges
- 5: Remove marks

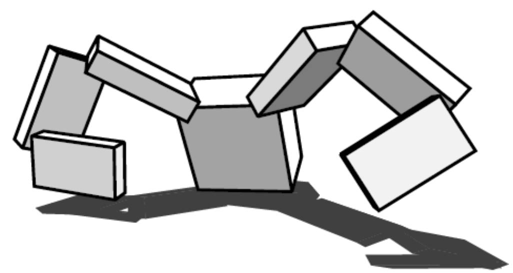




Directed graph for evolving virtual creatures



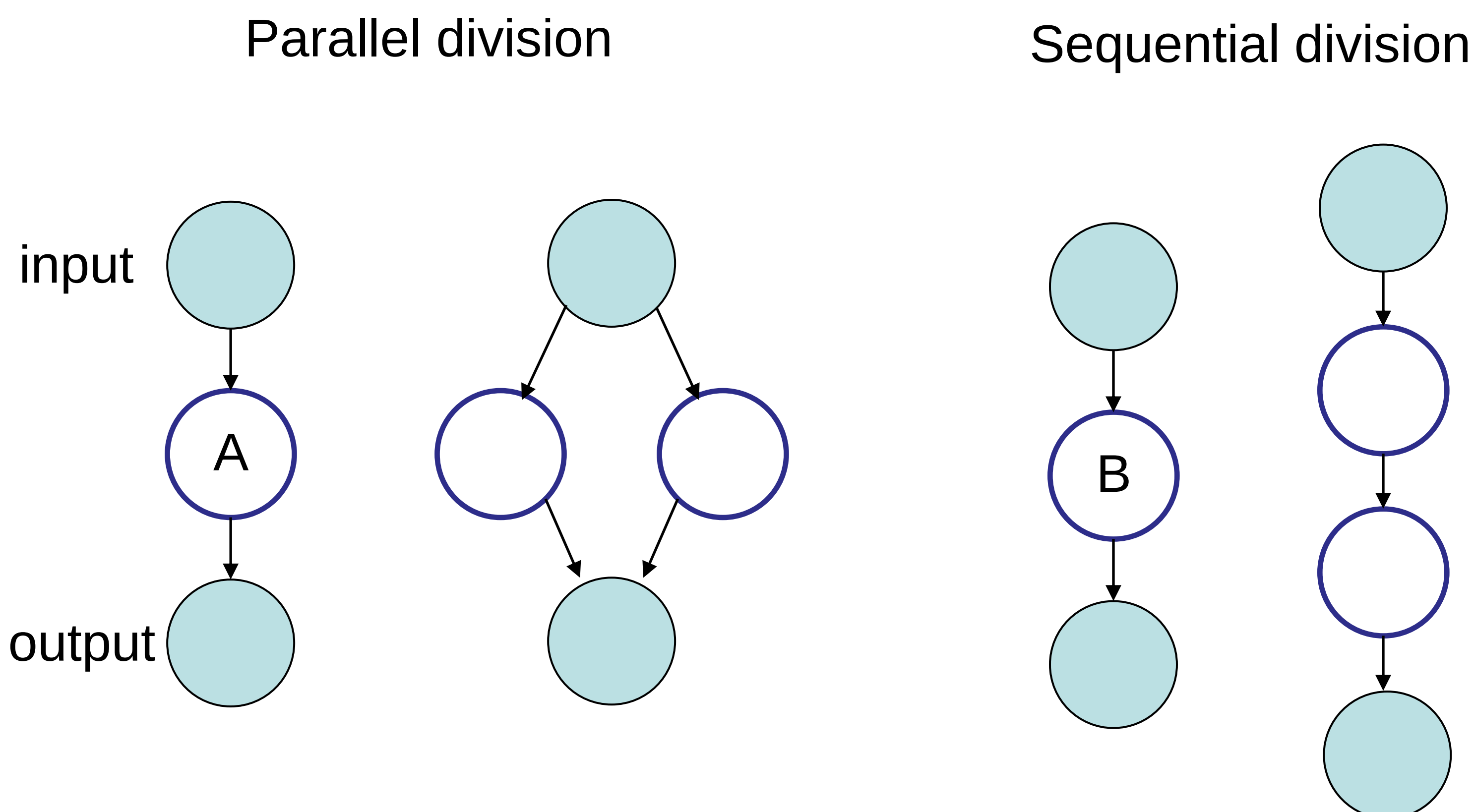




Karl Sims, 1994, Evolving Virtual Creatures

Cellular Encoding

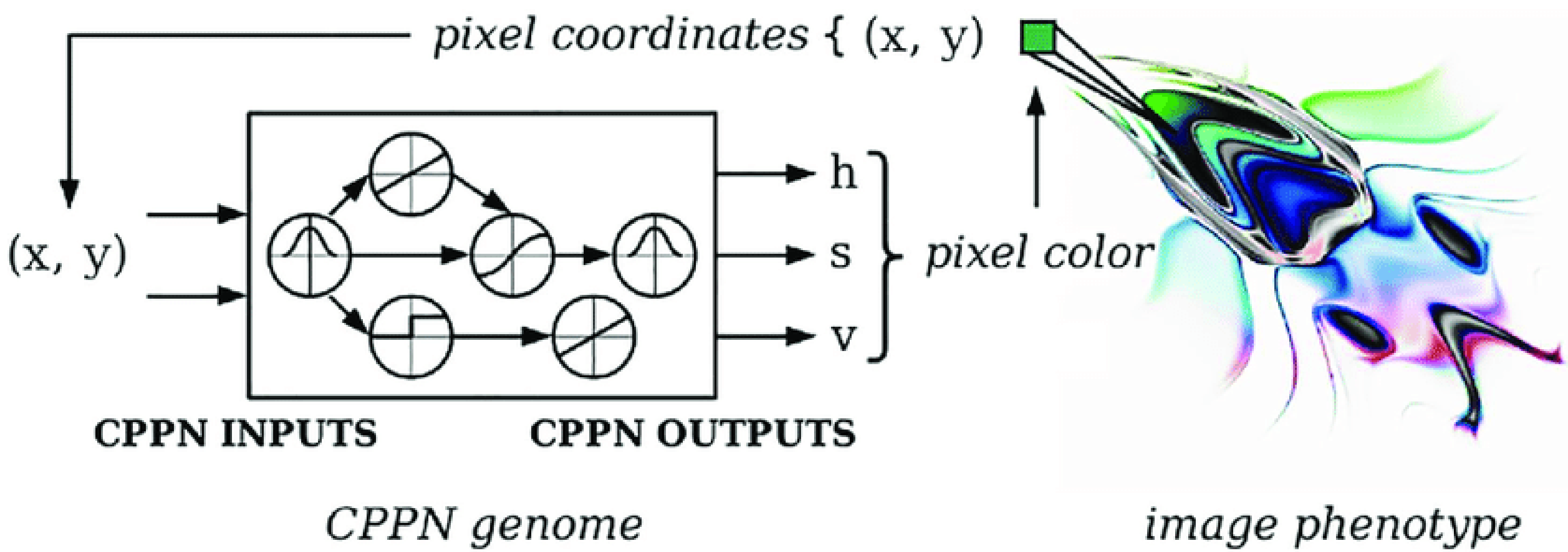
- Simulate cell division to create networks



Gruau (1994) Neural Network Synthesis Using Cellular Encoding And The Genetic Algorithm

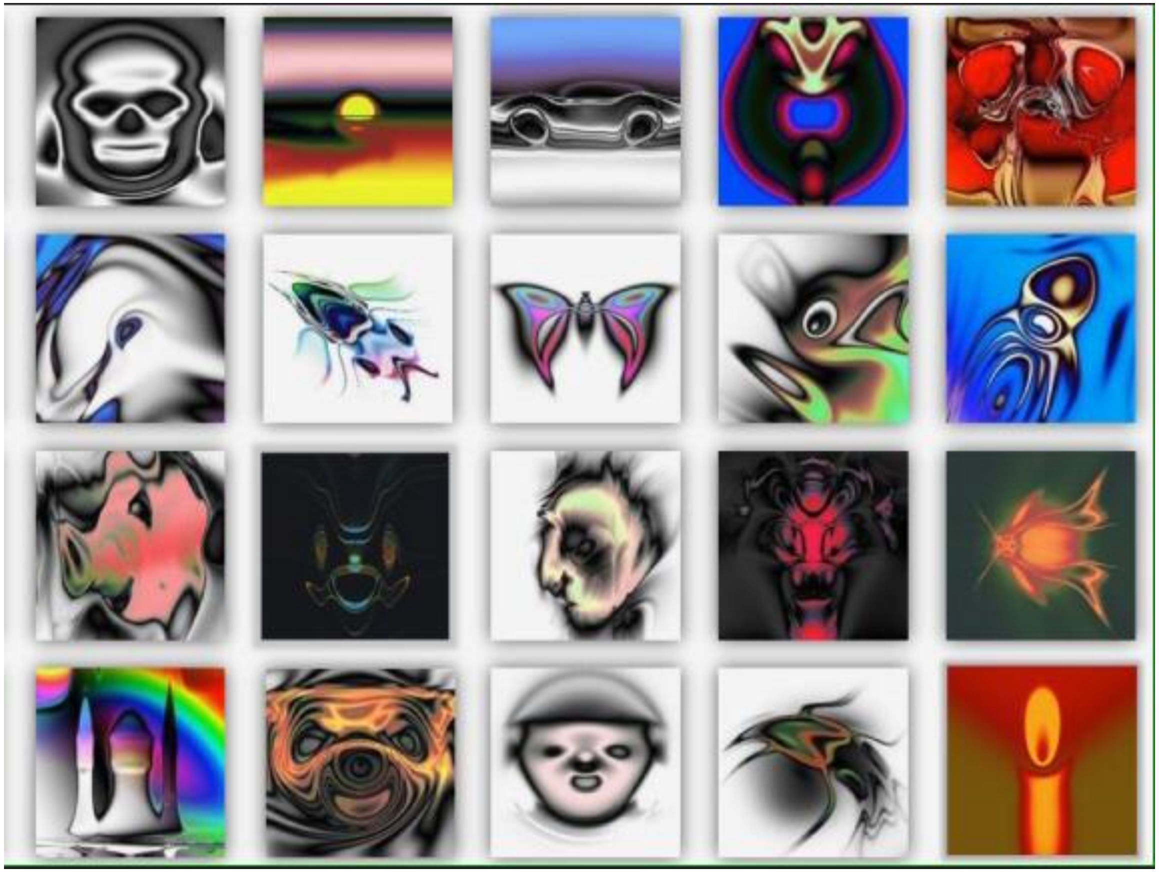
A Neural Network as a Developmental Abstraction?

- Compositional Pattern Producing Network (CPPN)



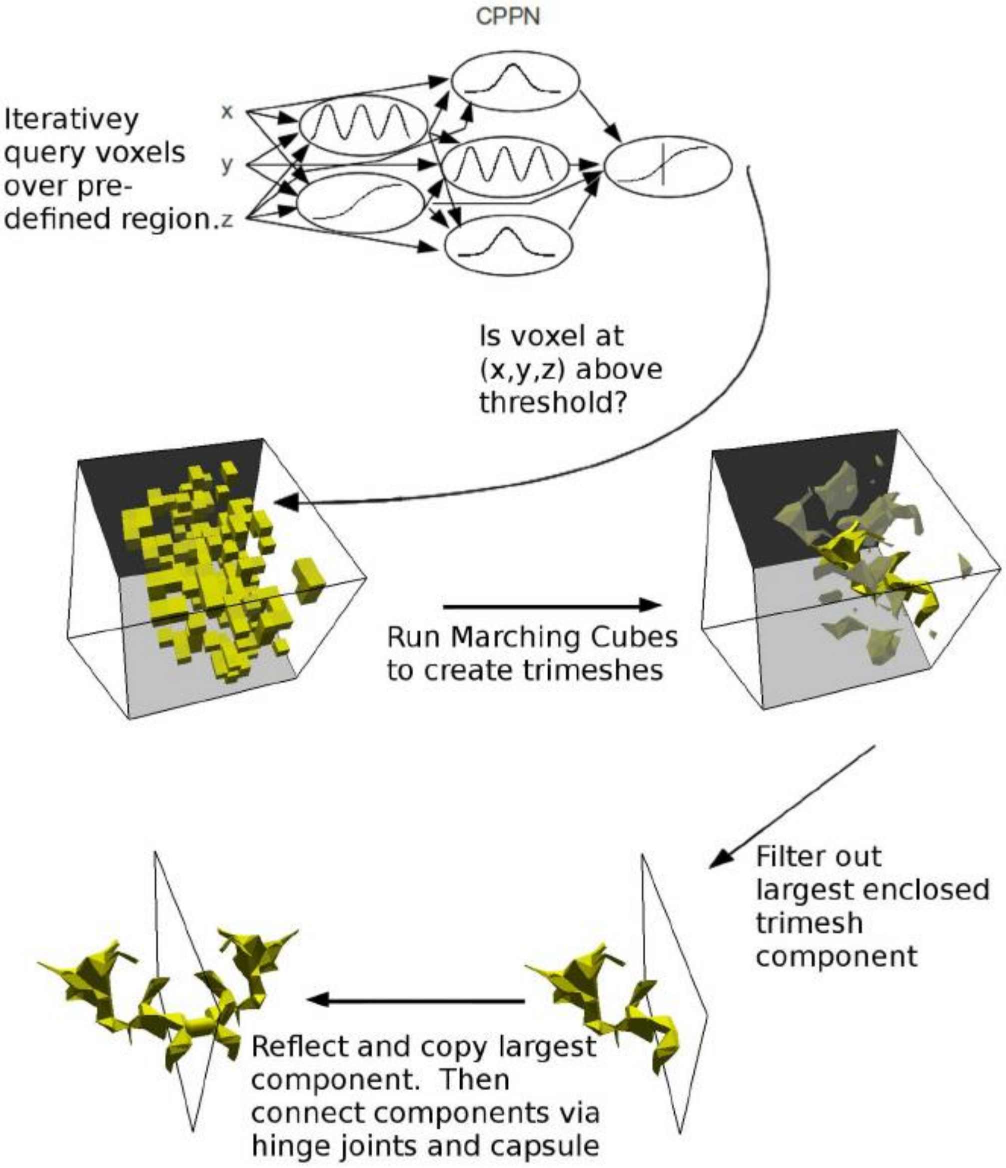
Helms and Clune, 2017, Improving HybrID: How to best combineindirect and direct encoding in evolutionary algorithms

A Neural Network as a Developmental Abstraction?



http://picbreeder.org/

Using CPPNs to create robots

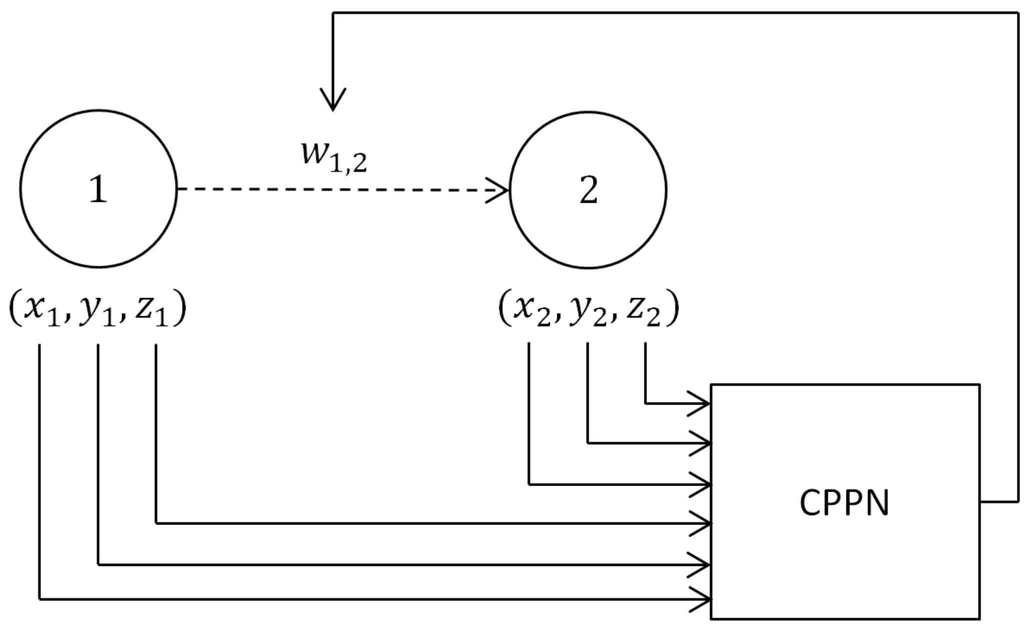


See this link for how CPPNs can be used to create soft-robots: https://www.youtube.com/watch?v=EXuR_soDnFo

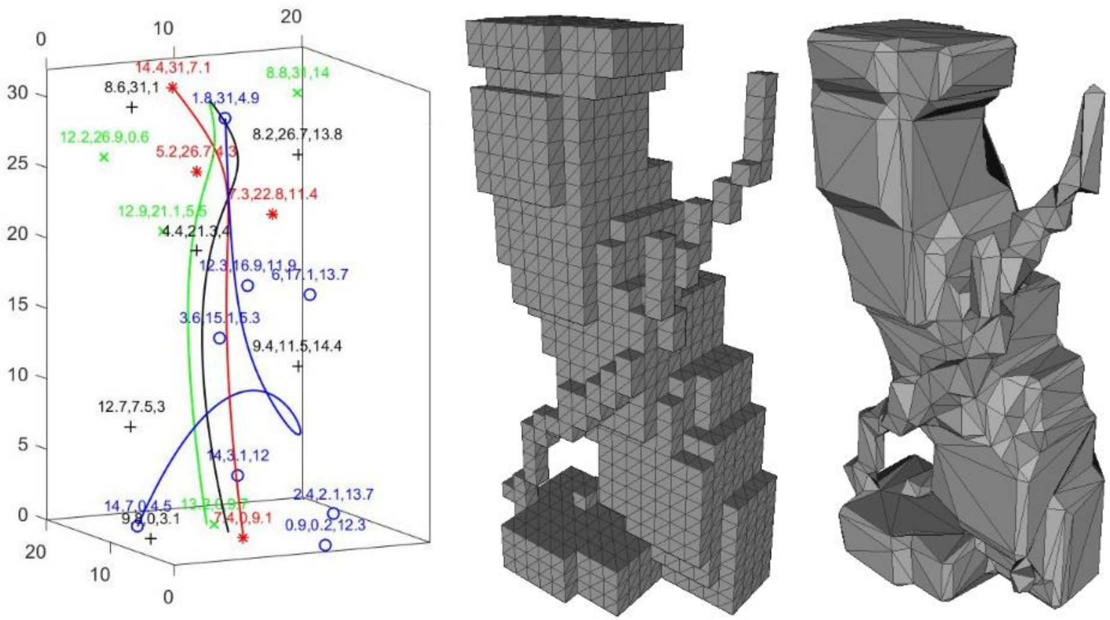
Auerbach JE, Bongard JC (2014) Environmental Influence on the Evolution of Morphological Complexity in Machines.

HyperNEAT

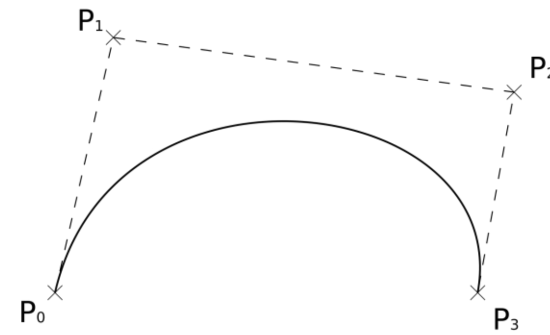
- Asks a CPPN about how assign weights to a neural network



Bezier Curve for Creating Robot Components



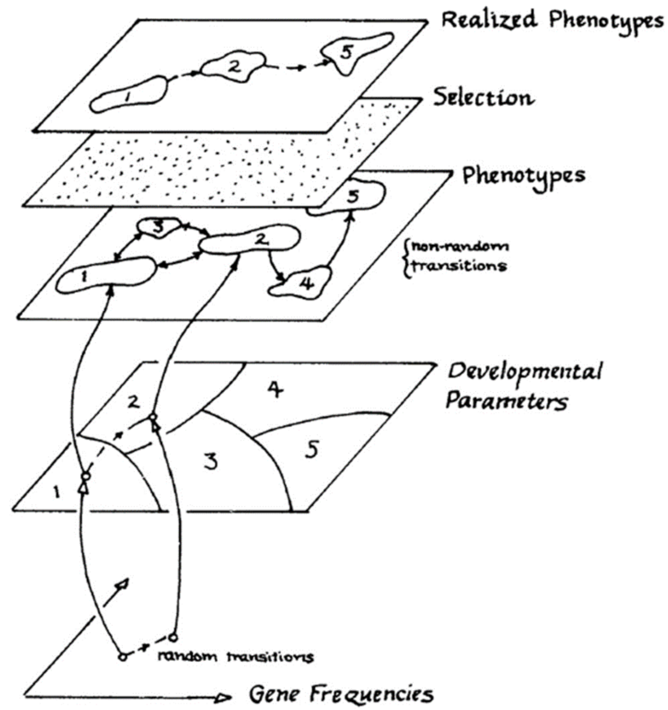
(a) (b) (c)



Collins et al., 2018, Towards the Targeted Environment-Specific Evolution of Robot Components

Developmental Systems

- Rewriting Systems
- Morphogens
- Neural Networks
- Bezier Curves



Swarm Intelligence

- Boids
- Particle Swarm Optimization
- Ant Colony Optimization
- Reconfigurable Robots

Swarm intelligence

- “The study of large collections of relatively simple agents that can collectively solve problems that are too complex for a single agent or that can display the robustness and adaptability to environmental variation displayed by biological agents”.
- Brain of brains (hive mind)
- Emergent intelligence forms
- Hive mind
- Bees dance
- How swarms can enhance the intelligence in groups

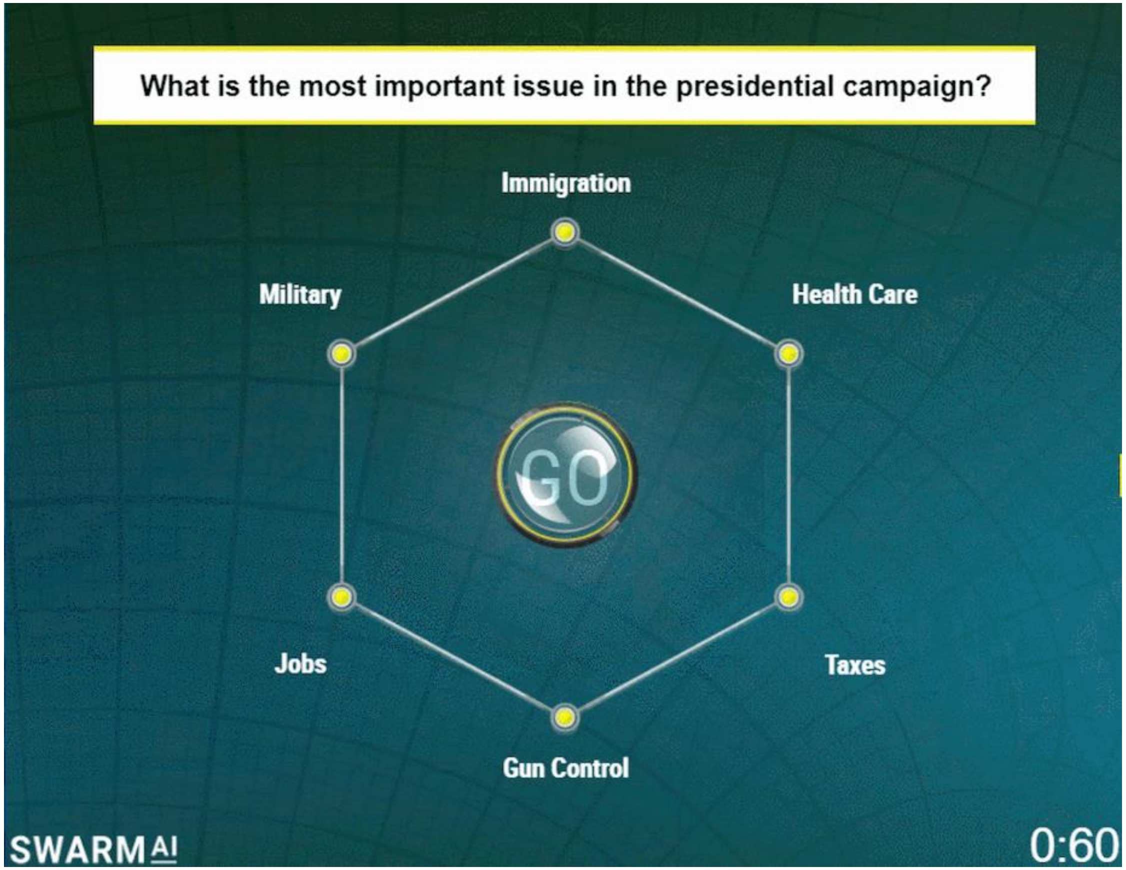
Floreano and Mattiussi (2008) Bio-inspired Artificial Intelligence

Swarm intelligence

- "The study of large collections of relatively simple agents that can **collectively solve problems** that are too complex for a single agent or that can display the robustness and adaptability to environmental variation displayed by biological agents".
- Brain of brains (hive mind)

Floreano and Mattiussi (2008) Bio-inspired Artificial Intelligence

Human swarm intelligence similar to ‘wisdom of the crowd’



Swarm Intelligence

- Emergent collective behavior:
  - Aggregation
  - Clustering
  - Foraging
  - Nest construction
- Extended phenotype:
  - How an animal’s genes can affect the world
  - E.g. is a beehive is an extended part of a bee?

Boids

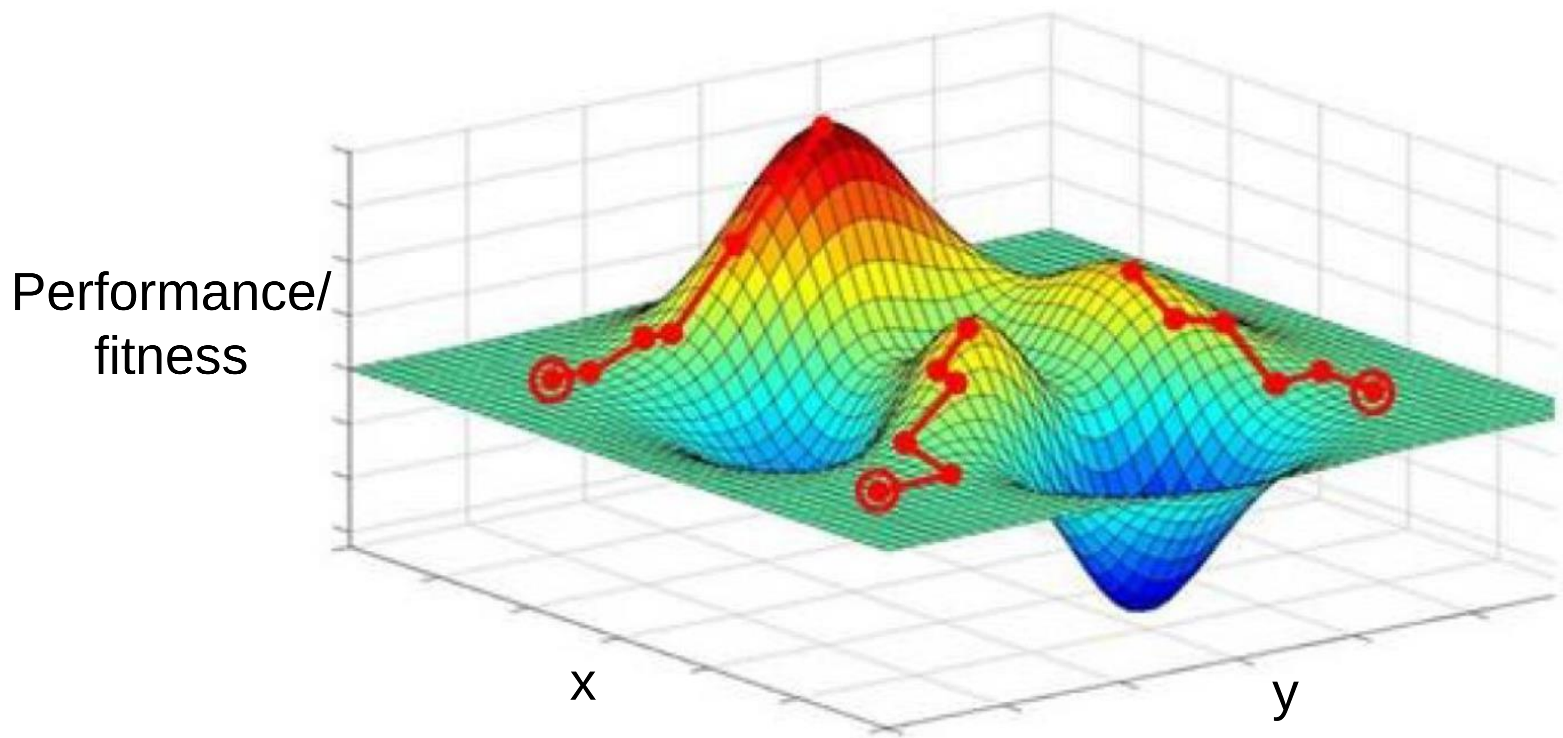
- Boid: ‘bird-oid object’
- Three rules:
  - **Separation**: boids steer away from close neighbors
  - **Alignment**: boids steer towards the average heading of their neighbors
  - **Cohesion**: boids steer towards the average position of its neighbors
- Exhibit aggregation
- Sebastian Lague: https://www.youtube.com/watch?v=bqtqltqcQhw

Reynolds, Craig W. (1987) Flocks, herds and schools: A distributed behavioral model

Particle Swarm Optimization (PSO)

- Population based metaheuristic like evolutionary algorithms
- Candidate solutions are *particles*
- A particle contains *positional* and *velocity* parameters
  - Position defines the adjustable/mutable parameters of an individual
  - Velocity represents how these parameters are updated
- The positional parameters can be viewed as the *genotype* of individuals.

Gradient ascent/descent of particles



Particle Swarm Optimization (PSO)

- (eq 1) updates positional parameters *x*
- (eq 2) updates velocity parameters *v*
  - *C1* and *C2* are *acceleration constants*
  - *pb* and *gb* are the *particle's best* and *global best* positions found

$$x_{i,d}(it+1) = x_{i,d}(it) + v_{i,d}(it+1) \quad (1)$$

$$\begin{aligned} v_{i,d}(it+1) &= v_{i,d}(it) \\ &+ C_1 * Rnd(0,1) * [pb_{i,d}(it) - x_{i,d}(it)] \\ &+ C_2 * Rnd(0,1) * [gb_d(it) - x_{i,d}(it)] \end{aligned} \quad (2)$$

Caption:

i particle's index, used as a particle identifier;

d dimension being considered, each particle has a position and a velocity for each dimension;

it iteration number, the algorithm is iterative;

$x_{i,d}$ position of particle i in dimension d;

$v_{i,d}$ velocity of particle i in dimension d;

C_1 acceleration constant for the cognitive component;

Rnd stochastic component of the algorithm, a random value between 0 and 1;

$pb_{i,d}$ the location in dimension d with the best fitness of all the visited locations in that dimension of particle i;

C_2 acceleration constant for the social component;

http://web.ist.utl.pt/gdgp/VA/pso.htm

Creating a PSO strategy

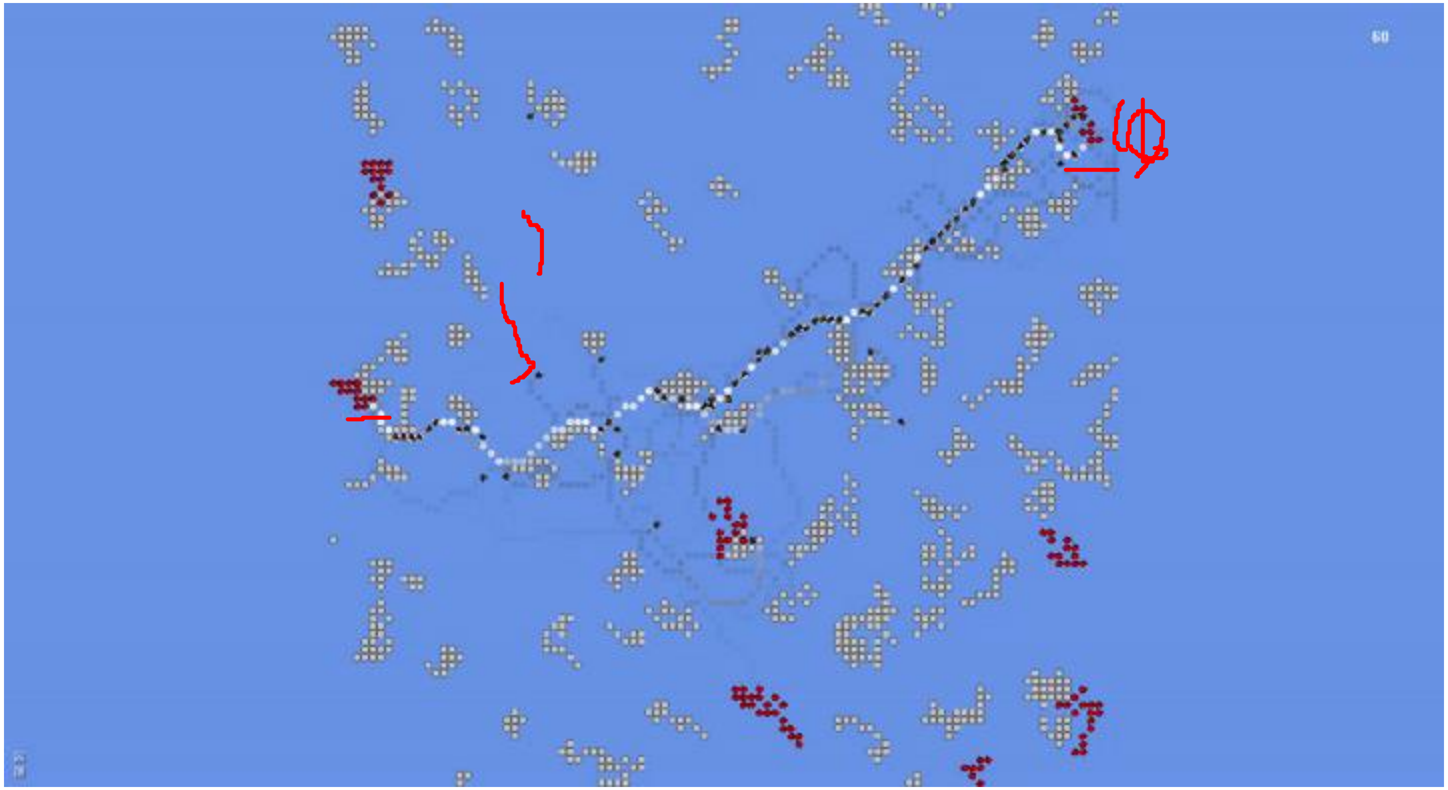
- See the PSO exercise online

$$x_{i,d}(it+1) = x_{i,d}(it) + v_{i,d}(it+1) \tag{1}$$

$$\begin{aligned} v_{i,d}(it+1) &= v_{i,d}(it) \\ &+ C_1 * Rnd(0,1) * [pb_{i,d}(it) - x_{i,d}(it)] \\ &+ C_2 * Rnd(0,1) * [gb_d(it) - x_{i,d}(it)] \end{aligned} \tag{2}$$

Ant Colony Optimization (ACO)

- **Stigmergy**: social communication through modification of the environment

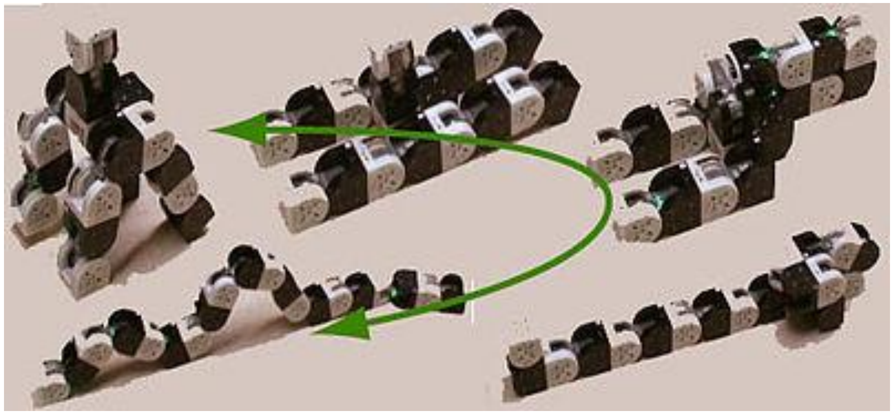

60

Why swarm systems?

- **Continuous adaptation**: dynamic network routing and urban transportation
- **Decentralized**, **Asynchronous**
- **Collective decision making**

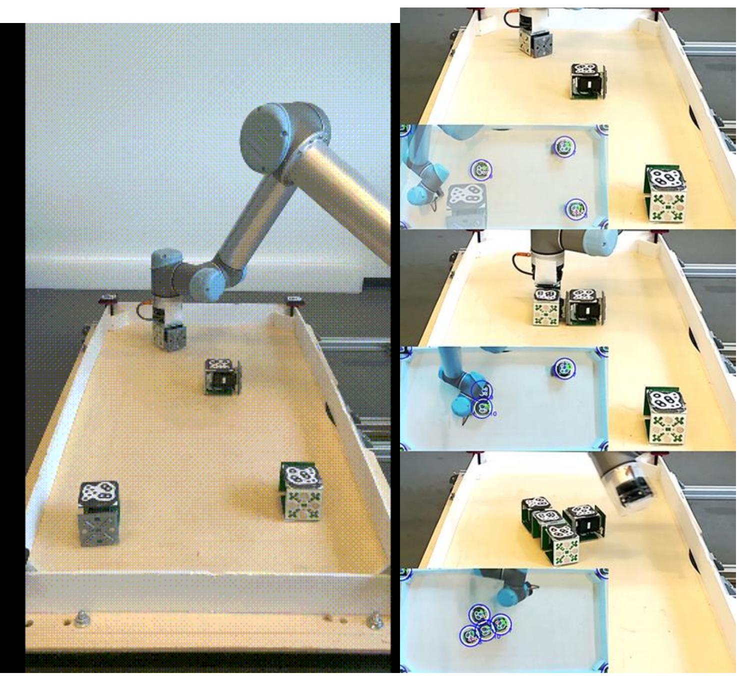
Reconfigurable robots

- Robots composed of modules that can change the shape of the robot to adjust its functionality
  - Possibly autonomously

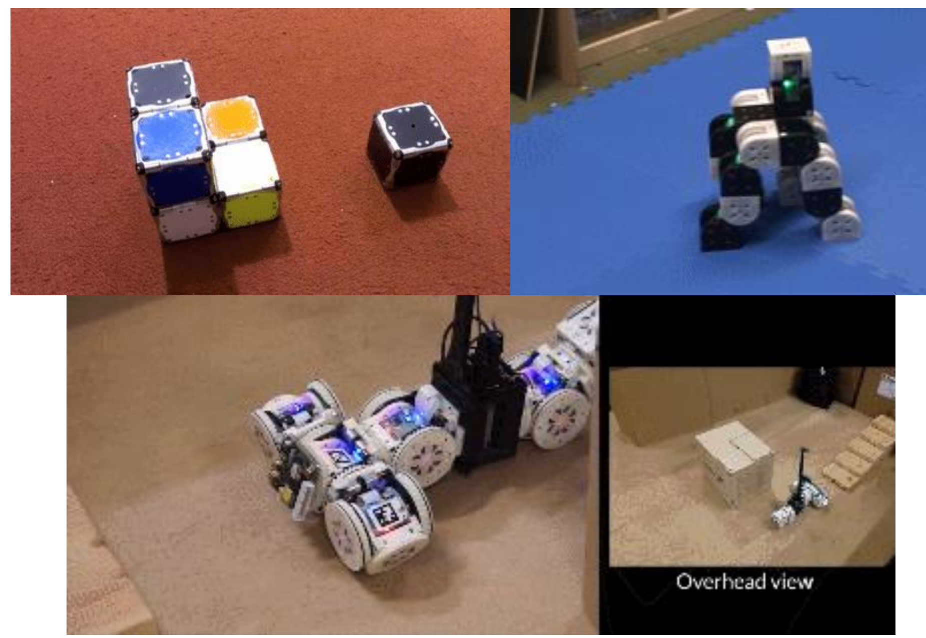


Reconfigurable modular robots

Passive Approaches

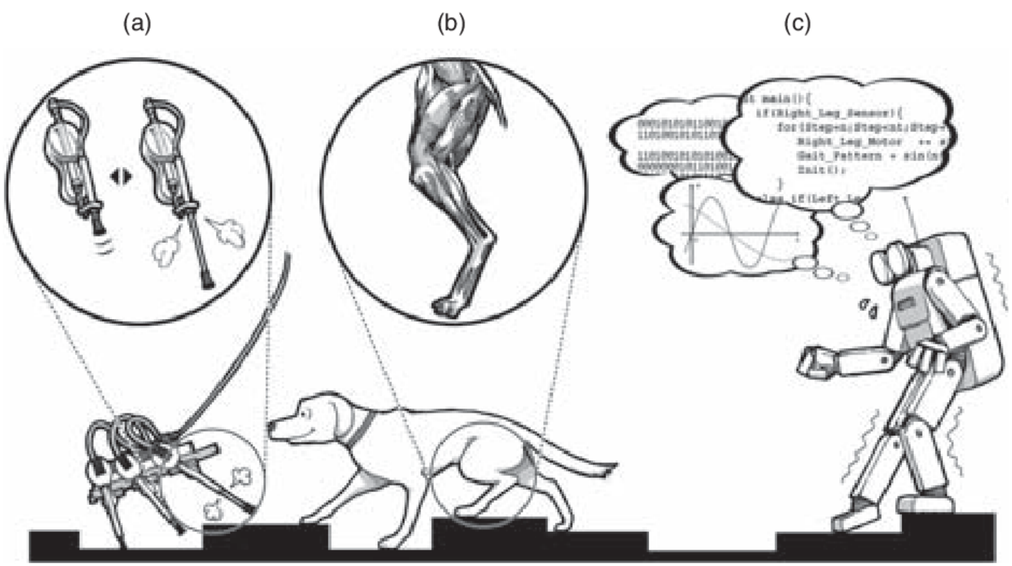


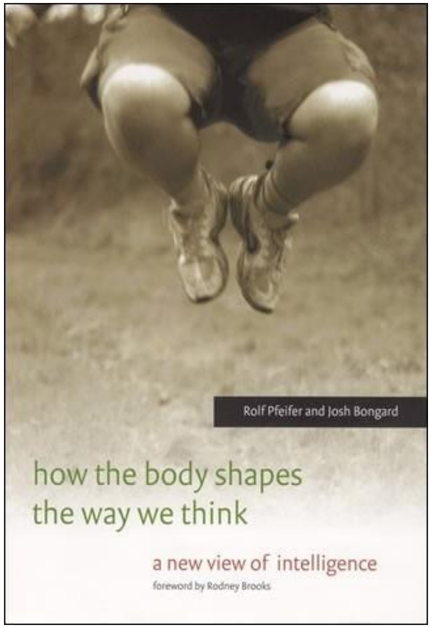
Active swarm approaches



Useful evolutionary robotics

- Shape-changing: evaluation different body shapes





Summary

- Boids
- Particle Swarm Optimization
- Ant Colony Optimization
- Reconfigurable Robots

Take home message

- Design artificial systems considering the genotype to phenotype mapping
  - Compact storage
  - Reuse of information
  - Modularity
  - Decentralized controllers