

## Open-ended Neuroevolution

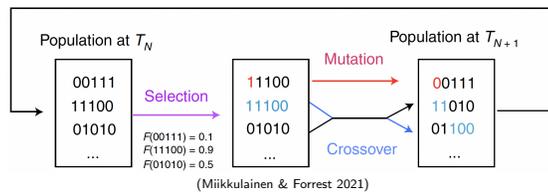
Risto Miikkulainen

February 21, 2026

Navigation icons

### 1. Standard approach to computational evolution

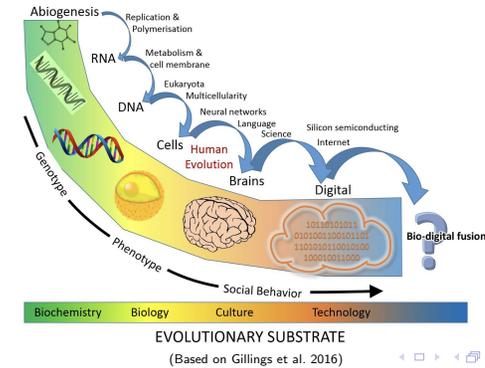
- ▶ Evolutionary computation often uses strong selection for fast optimization.
- ▶ Small populations and engineered operations lead to quick convergence.
- ▶ This approach works for well-defined problems but may miss solutions.
- ▶ Engineers are highly impatient!



Navigation icons

## Open-ended Discovery of Complex Behavior

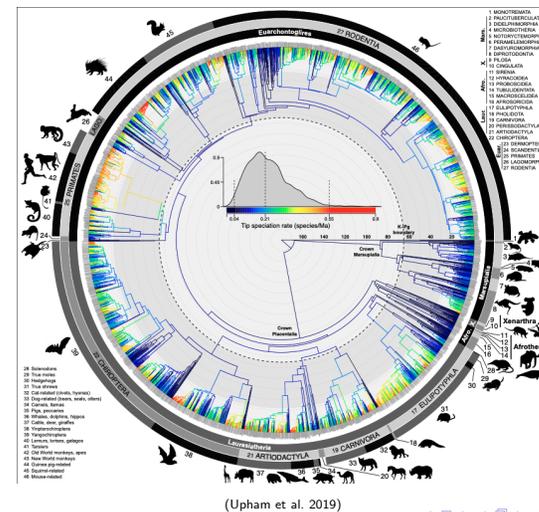
- ▶ Neuroevolution can find successful behavior for given tasks.
- ▶ Biologicity is open-ended: Operates continuously, without specific goals.
- ▶ Elements of open-endedness could lead more powerful neuroevolution:
  1. Neutrality with weak selection
  2. Enhanced exploration through extinction events
  3. Evolvable representations
  4. Expressive (not just indirect) encodings.
  5. Major transitions in complexity
  6. Co-evolution of body and brain
  7. Co-evolution of brain and static environments.
  8. Co-evolution of brain and dynamic environments.



Navigation icons

### Biological Approach

- ▶ Biological evolution involves large populations and weak selection.
- ▶ Many mutations are neutral and do not affect fitness.
- ▶ Diversity stays in the population even when not immediately beneficial.
- ▶ Biology has a lot of patience :-)



Navigation icons

## Opportunity with Increased Computational Power

- ▶ Impatience because of limited computational power.
- ▶ With billionfold increases, new approaches are possible.
  - ▶ Large populations, neutral mutations, deep time, large simulations.
  - ▶ Develop foundation models for neuroevolution?
- ▶ Computational scaleup led to deep learning and GenAI; similar advances possible for neuroevolution as well.



(iStock 2024)



## 2. Extinction Events in Biological Evolution

- ▶ Five large-scale extinction events have altered the course of evolution.
- ▶ Example: The Cretaceous-Paleogene event displaced dinosaurs with mammals.
- ▶ Question: Are extinction events accidents or do they serve a role in evolution?
- ▶ Extinctions may reset evolution, favoring higher evolvability and complexity.

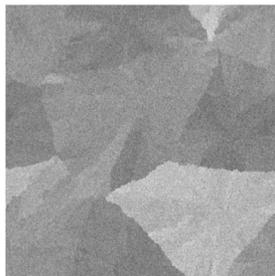


(DALL-E 2024)

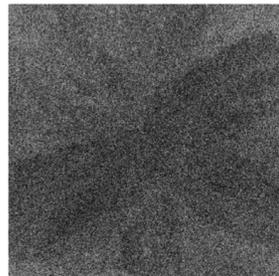


## Computational Experiments on Extinctions

- ▶ Extinction events can be studied in an abstract setting:
  - ▶ Individuals encoded with a 2D location/niche and evolvability.
  - ▶ Evolvability determines how likely the offspring is to occupy another niche through mutation.
- ▶ Starting from the center, population evolves until all niches are filled.
- ▶ Further evolution only through drift; evolvability stagnates.



(a) Control - 2,000 Gens.



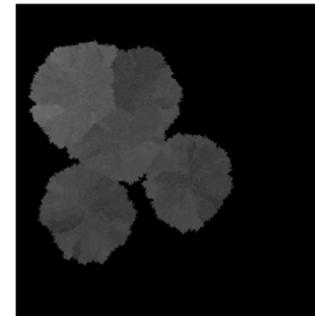
(b) Control - 15,000 Gens.

(Lehman & Miikkulainen 2015)  
Lighter=more evolvable

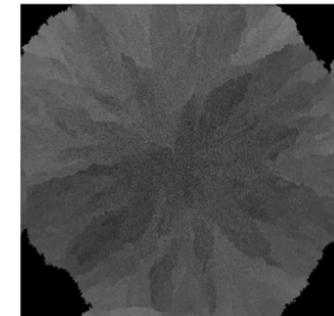


## Evolvability Through Extinctions

- ▶ Extinction events remove most of the population, leaving a few individuals.
- ▶ The ones with higher evolvability fill more niches.
- ▶ Repeated extinction thus favors individuals with higher evolvability.
- ▶ Extinction events accelerate evolution!



Soon after extinction



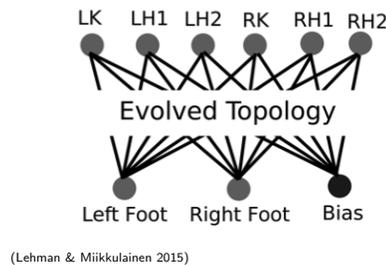
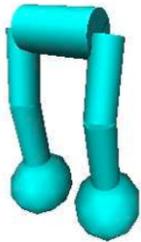
Later after extinction

(Lehman & Miikkulainen 2015)



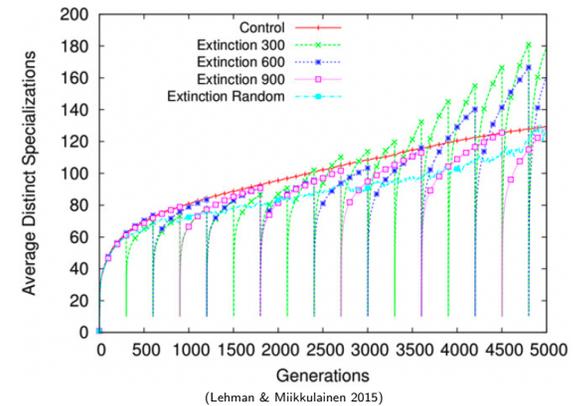
## Extinction in the Bipedal Walker Domain

- ▶ Extinction events can affect behavioral evolution as well.
- ▶ E.g. Bipedal walker:
- ▶ Input: Whether each foot touches the ground.
- ▶ Output: motors on left and right knee and hip.
- ▶ Nice: Where does it end up?



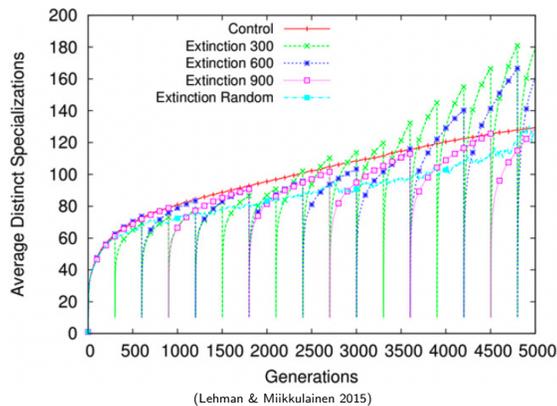
## Extinction in the Bipedal Walker Domain

- ▶ Extinction events lead to faster rebounds.
- ▶ After each extinction, the population fills more niches than before.
- ▶ Extinction accelerates evolution and increases novelty in solutions.



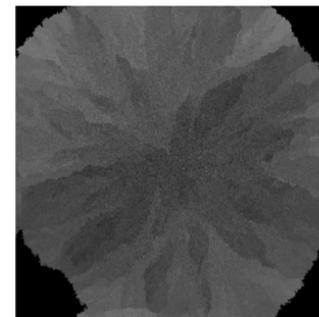
## Conclusion on Extinction Events

- ▶ Extinction events have the potential to reset evolution and favor more evolvable solutions.
- ▶ Even at smaller scales, extinction events may be beneficial for long-term innovation.
- ▶ Combining extinction events with large populations and weak selection could accelerate opened evolution in neuroevolution systems.



## 3. Evolvable Representations

- ▶ Evolutionary algorithms often focus on creating and maintaining diversity in genotypic and phenotypic space (quantity)
- ▶ Evolvability makes it possible to create larger changes faster (quality).
  - ▶ The main idea is to adapt the genotype-phenotype mapping.
  - ▶ Indirect encodings are crucial to supporting evolvability.
- ▶ Both of these approaches can lead to better high-fitness solutions.

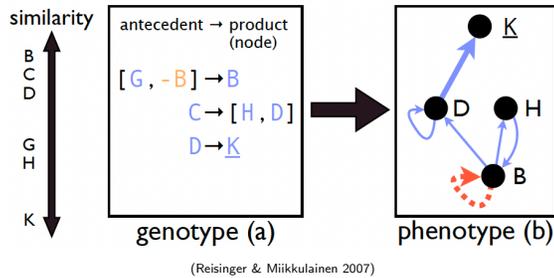


(Lehman & Miikkulainen 2015)



## GRNs Can Support Evolvability

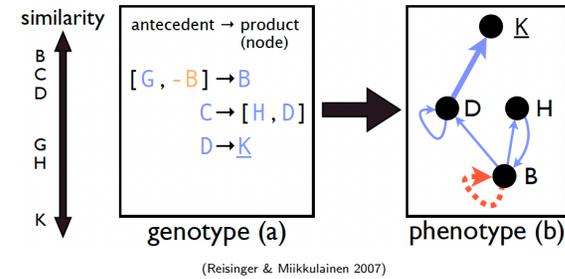
- ▶ Genetic Regulatory Networks (GRNs) are indirect encodings that evolve over time.
- ▶ GRNs encode rules for building neural networks.
- ▶ Products in the GRN create nodes and connections between them.
- ▶ Evolvability emerges through continuous mutations in the rules.



◀ ▶ ⏪ ⏩ 🔍 🔄

## Constructing neural networks from GRNs

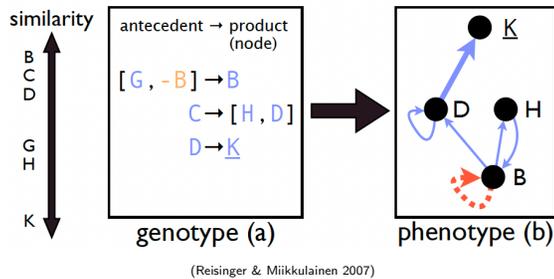
- ▶ Each rule has a regulatory(antecedent) and transcription(product) region.
- ▶ Variables are regulatory factors (proteins) with tolerance (similarity)
- ▶ When existing products match antecedents within tolerance;
  1. Products are generated as nodes.
  2. Antecedent nodes are connected to product nodes.



◀ ▶ ⏪ ⏩ 🔍 🔄

## GRN → Neural Net Example

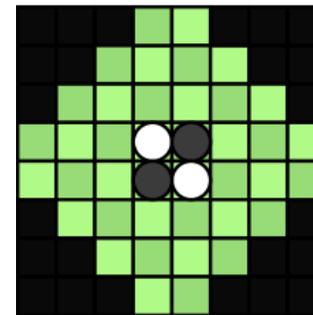
- ▶ Starting with G and no B, B is created (as a node)
  - ▶ With input from G (not shown).
  - ▶ With an inhibitory link  $B \rightarrow B$ .
- ▶ B is close to C: H and D are created (as nodes)
  - ▶ With connections from B.
  - ▶ H is close to G: connection from H to B.
  - ▶ D is close to C: connection from D to itself.
  - ▶ D is close to C: connection from D to H (not shown).
- ▶ D creates K, the output node
  - ▶ With a connection from D.



◀ ▶ ⏪ ⏩ 🔍 🔄

## Test Domain: Nothello

- ▶ GRN-based neuroevolution was tested in Nothello (simplified Othello):
  - ▶ Still: Take turns to place pieces (black/white)
  - ▶ Still: Sandwiched pieces are flipped.
  - ▶ But: Smaller diamond-shaped board.
  - ▶ But: Player with the fewest pieces wins.
- ▶ Simpler, faster, but still challenging.

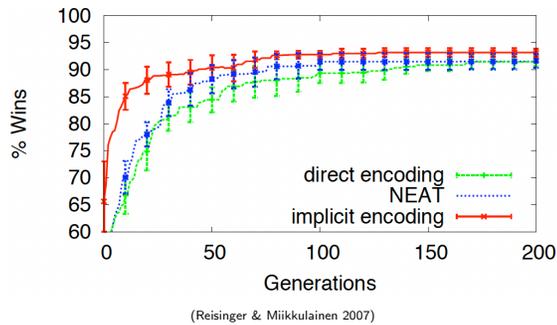


(Reisinger & Miikkulainen 2007)

◀ ▶ ⏪ ⏩ 🔍 🔄

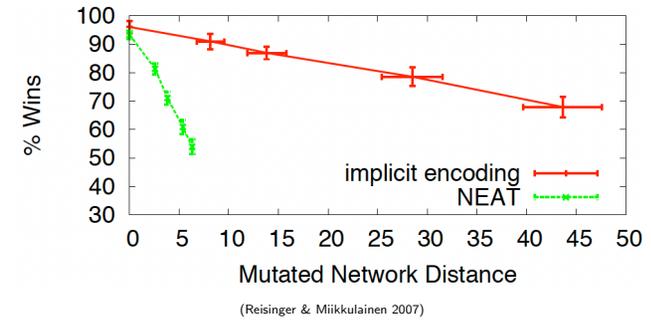
## Performance in the Nothello Domain

- ▶ Neural networks evolved as heuristic evaluators.
- ▶ Coevolution process: Each network evaluated against others in the population.
- ▶ The GRN-based indirect encoding evolved better strategies faster than evolution of feature weights and NEAT.



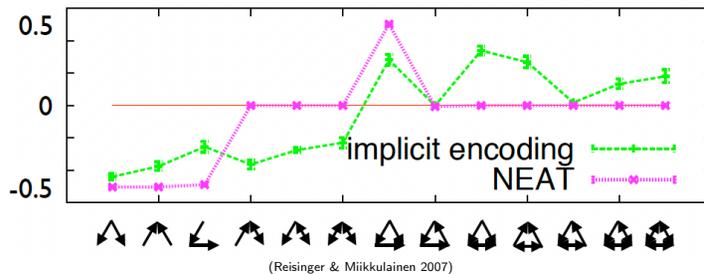
## Performance Arising from Evolvability

- ▶ In coevolution, the fitness function changes continuously.
- ▶ High evolvability is thus a major advantage in coevolution.
- ▶ Evolvability is measured as the average fitness of mutated offspring.
- ▶ The GRN-based encoding results in more robust mutations compared to NEAT.



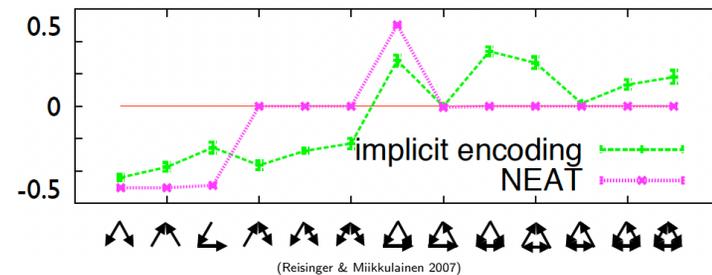
## Network Motifs with High Evolvability

- ▶ The GRN-based approach discovers a wider variety of network motifs.
- ▶ Recurrent motifs are common in GRN networks, supporting robust evolvability.
- ▶ This leads to more complex behaviors and open-ended discovery.



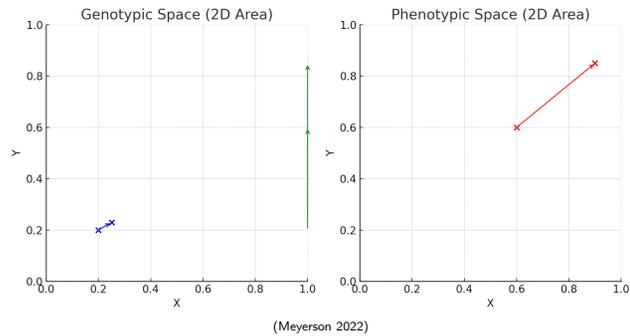
## Conclusion on Evolvable Representations

- ▶ Evolvable representations support long-term evolution and adaptation to changing environments.
- ▶ Indirect encodings like GRNs enable evolvability through soft, continuous mutations.
- ▶ Evolvability is especially useful in coevolution where the fitness function changes continuously.
- ▶ Evolvability is thus essential for open-ended discovery of increasingly complex solutions.



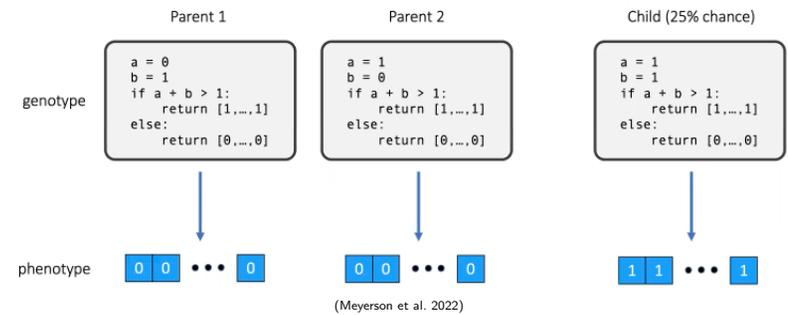
## 4. Expressive Encodings Overview

- ▶ Expressive encodings allow for large jumps in the search space.
- ▶ In particular, small changes in the genotype can result in large changes in the phenotype (miracle jumps).
- ▶ Standard evolutionary algorithms with direct encodings cannot easily make such jumps.
- ▶ Expressive encodings are highly evolvable and open-ended.



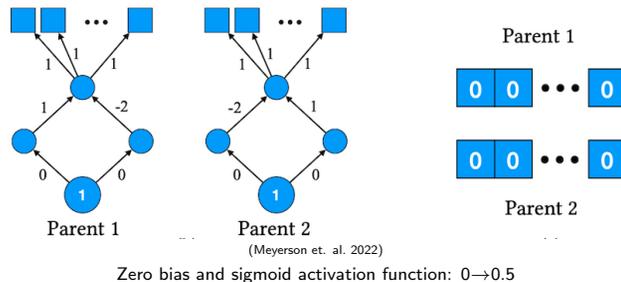
## Example of Miracle Jumps: GP

- ▶ Genetic programming (GP) can create miracle jumps through conditionals or specific segments.
- ▶ E.g. two GP parent phenotypes are all-0s.
- ▶ But there's a 25% chance that their crossover is all-1s.



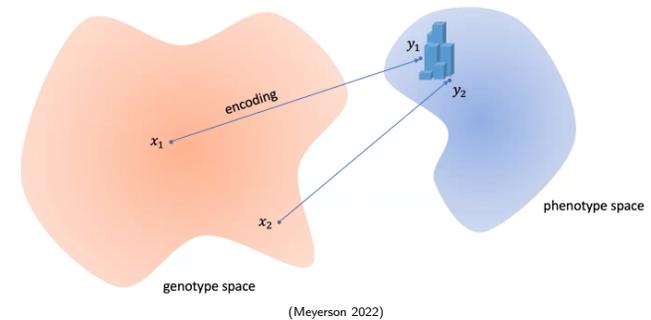
## Example of Miracle Jumps: NE

- ▶ Neuroevolution (NE) can also produce miracle jumps.
- ▶ E.g. neural networks evolved to produce a bit string.
- ▶ The two parents differ only in the weights of the second layer.
- ▶ With uniform crossover, 25% chance of jumping from all-0s to all-1s.
- ▶ Direct encoding cannot achieve such jumps as effectively.



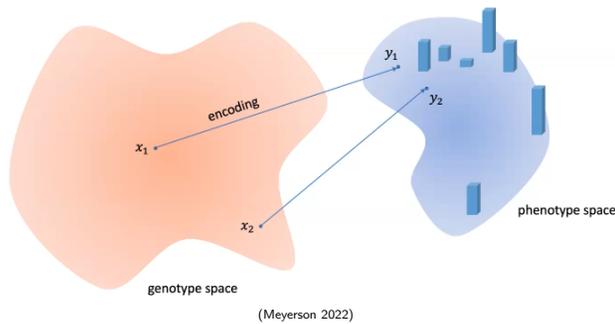
## Power of Expressive Encodings

- ▶ In a standard crossover, offspring phenotype is somewhere in between the parent phenotypes.



## Power of Expressive Encodings

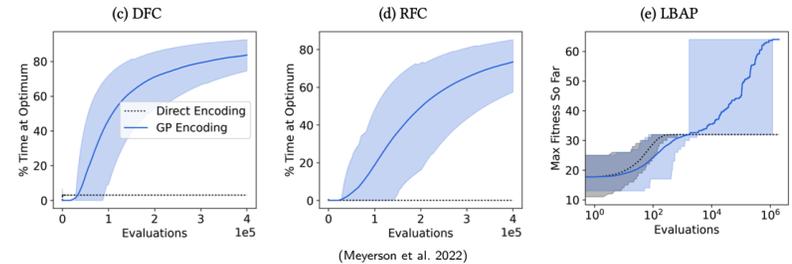
- ▶ With Expressive Encodings, the offspring phenotypes can be anywhere in the space.
- ▶ Even with a simple crossover, recombination samples from an arbitrary distribution of child phenotypes.



Navigation icons: back, forward, search, etc.

## Power of Expressive Encodings

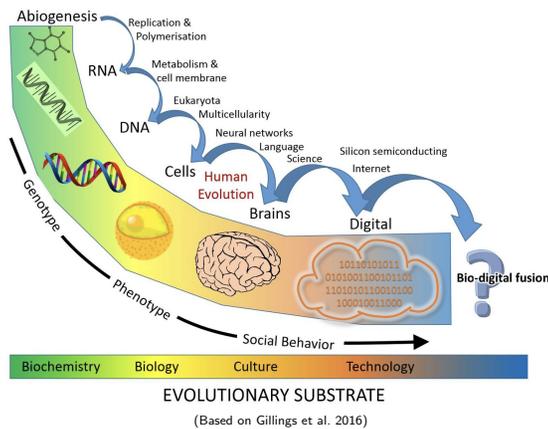
- ▶ Expressive encodings outperform direct encodings theoretically and experimentally (Deterministic Flipping, Random Flipping, Large Block Assembly).
  - ▶ Even with flat landscape, break through eventually.
- ▶ Much of the genetic structure is shared, similar to biological systems.
- ▶ A good idea to use GP or NN as a representation; best chance for open-ended evolution!



Navigation icons: back, forward, search, etc.

## 5. Major Transitions

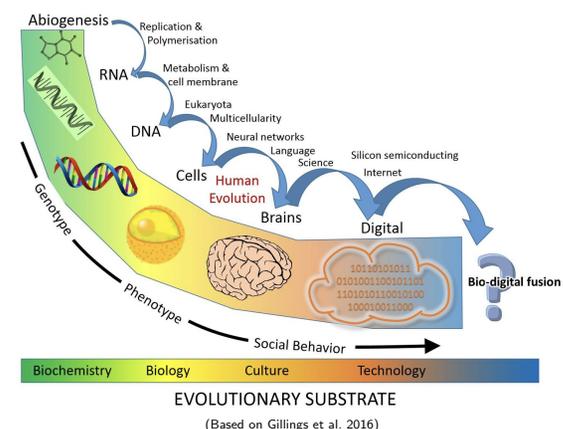
- ▶ Open-ended discovery may require major transitions in complexity.
- ▶ Biological evolution progressed through several transitions, e.g.:
  - ▶ From self-replicating molecules to chromosomes.
  - ▶ From single cells to multicellular organisms.
  - ▶ Formation of eusocial societies and language.
- ▶ Each transition results in more complex individuals and cooperative roles.



Navigation icons: back, forward, search, etc.

## Challenges in Establishing Major Transitions

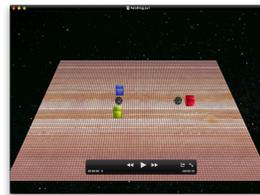
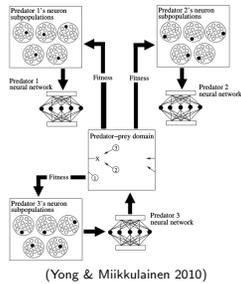
- ▶ Not fully understood in biology.
- ▶ Key questions:
  - ▶ How do individuals specialize and lose the ability to reproduce independently?
  - ▶ Are there multiple levels of selection or just one?
- ▶ Computational studies can help, but simulating major transitions in behavior is challenging.



Navigation icons: back, forward, search, etc.

## Mechanisms for Major Transitions

- ▶ Ingredients for major transitions exist in computational experiments:
  - ▶ Evolving cooperative structures (e.g., Hierarchical ESP).
  - ▶ Agents communicating and coordinating to achieve shared goals.
- Also (to be discussed next):
  - ▶ Coevolution of body and brain.
  - ▶ Emergence of new challenges through competitive processes (environment, adversaries).
- ▶ However, actual transitions have not been discovered in computation yet.



(<https://nn.cs.utexas.edu/?armsrace>)