Neuroevolution Tutorial

Artificial Life Conference 2024

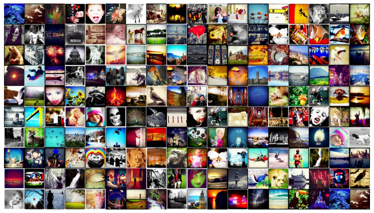
Risto, Sebastian, Yujin

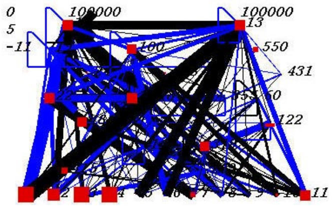
Outline

1. Motivation

From imitation to creativity
Exploration vs. refinement (Evo vs. RL)

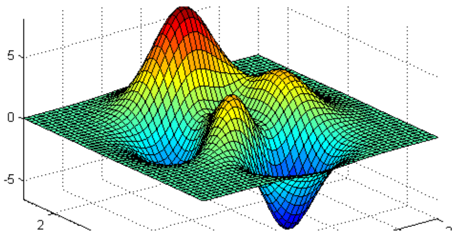
1. Neuroevolution in Alife: From Imitation to Creativity





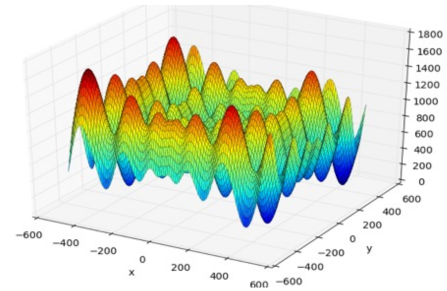
- Much of AI focuses on imitation
    - I.e. gradient descent on labeled datasets
    - Powerful in prediction: object recognition, diagnosis, forecasting, etc.
- Alife focuses on behavior
    - Gradients not available
    - Needs to be discovered
- How can we create novel behaviors?

Reinforcement Learning is One Approach



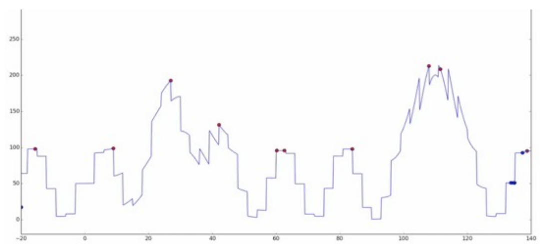
- Approximating gradient descent
  - Explore around the current solution
  - Improve it gradually
  - Can climb the nearest hill well

...but Creativity in RL is Limited



- Space is too large
  - Multiple starts won't help
- Space is too high-dimensional
  - Little improvement from one step
- Space is deceptive
  - Can only find the nearest hill

Solution: Evolutionary AI



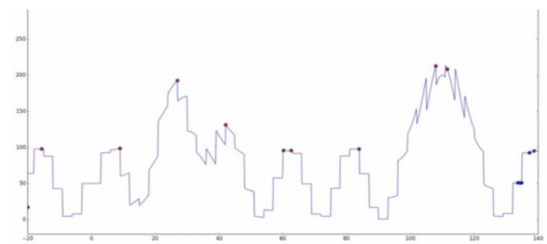
Based on population-based search

- Many individuals spread out, sharing information
- Not limited to differentiable domains: configurations, choices ok

Not limited to incremental improvement

- Large jumps possible, can be more creative

Scaling up Evolutionary AI



Works in large scales

- Structured search works in large spaces (e.g. 2^2^70; Hodjat & Shahzad 2016)
- Multiple variables optimized at once (e.g. up to 1B; Deb et al. 2017)
- Multiple objectives and novelty get around deception (Shahrzad and Hodjat 2020)

Neuroevolution uses population-based search to discover behavior

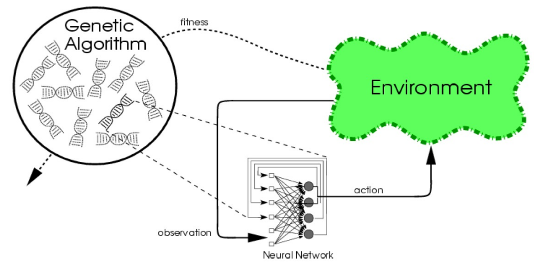
- A useful approach to Alife

2. Basics

2.1 Evolution Basics ®
- population, selection, variation, search

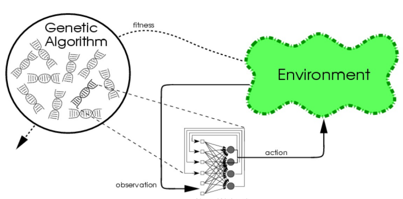
2.2 Neuroevolution Basics
- Evolving bipedal walker
- NEAT

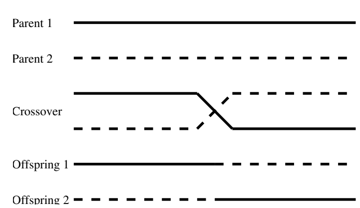
2.1 Evolution Basics: Encoding, Evaluation, and Selection



- A population of encodings (e.g. lists or trees)
- Decoded into individuals that are evaluated in the domain
- Good individuals retained, bad thrown away

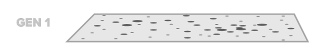
Creating Variation



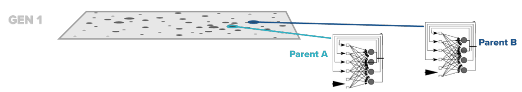


- New individuals generated from the parent encodings
  - Crossover: combine building blocks from two parents
  - Mutation: create new building blocks

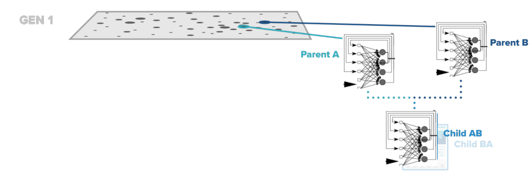
Population-based Search



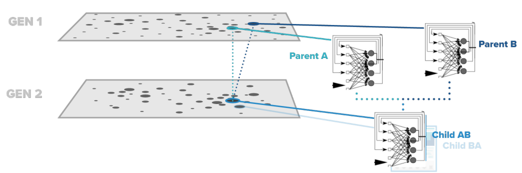
Population-based Search



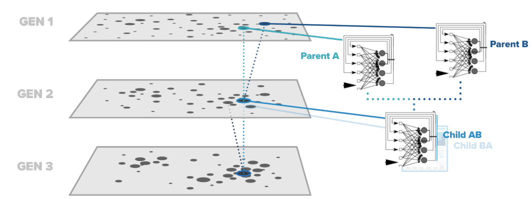
Population-based Search



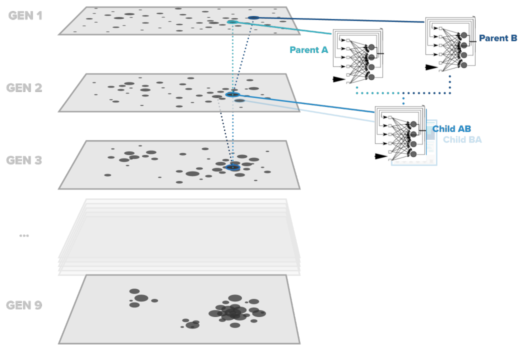
Population-based Search



Population-based Search

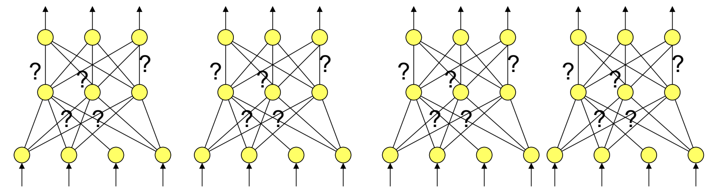


Population-based Search

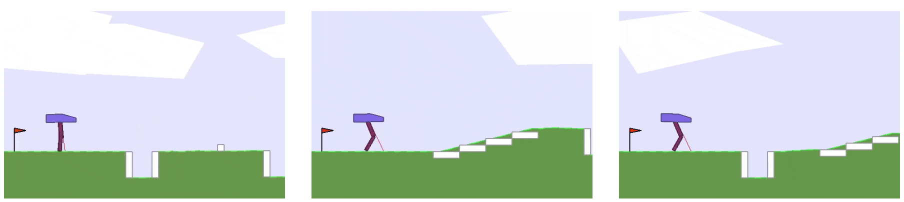


2.2 Neuroevolution Basics

- Genome is a direct encoding
- Genes represent a vector of weights
- NE (using GA, ES, etc.) optimizes the weights for the task



Learning to Walk

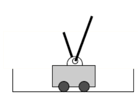


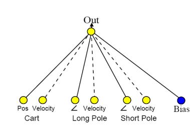
Demo

https://blog.otoro.net/2017/11/12/evolving-stable-strategies/

Neuroevolution of Augmenting Topologies (NEAT)

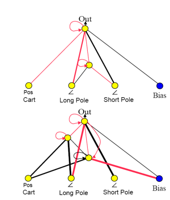
- *Historical markings* match up different structures
- Speciation
  - Keeps incompatible networks apart
  - Protects innovation
- Incremental growth from minimal structure, i.e. *complexification*
  - Avoids searching in unnecessarily high-d space
  - Makes finding high-d solutions possible





Method	Evaluations
Gruau et al. 1996 Cellular Encoding	840,000
Gomez and Miikkulainen 1999 ESP	169,466
NEAT	33,184

20 trials



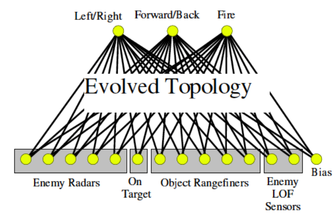
Stanley, K. O., & Miikkulainen, R. (2002). Evolving neural networks through augmenting topologies. *Evolutionary computation*, *10*(2), 99-127.

3.1 Indirect encodings

1 Evolution + learning ®
- Genetic Regulatory Networks
- Learning from *prediction, *parents, *others in the population
- *Baldwin effect, *Lamarckian evolution, epigenetics
- *Rethinking innateness: evolving pattern generators

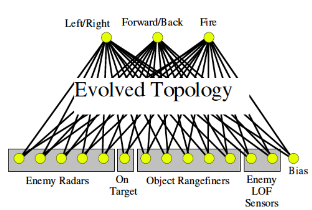
2 Generative Encodings

Combining Learning and Evolution



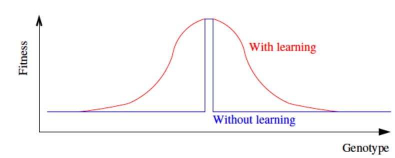
- Good learning algorithms exist for NN
  - Why not use them as well?
- Evolution provides structure and initial weights
- Fine tune the weights by learning

Lamarckian Evolution



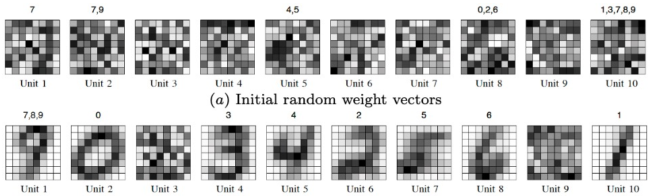
- Lamarckian evolution is possible[8,32]
  - Coding weight changes back to chromosome
- Difficult to make it work
  - Diversity reduced; progress stagnates

Baldwin Effect



- Learning can guide Darwinian evolution as well[5,32,34]
  - Makes fitness evaluations more accurate
- With learning, more likely to find the optimum if close
- Can select between good and bad individuals better
  - Lamarckian not necessary

Synergy of Evolution and Learning



(a) Initial random weight vectors

(b) Final competitive learning weight vectors

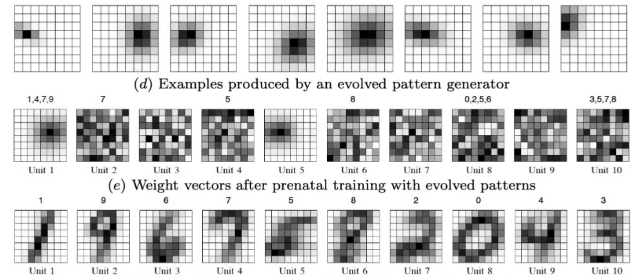
If learning is available, evolution will use it
Deep synergy: Prenatal learning through internally generated patterns

- Evolve a starting point, learning mechanism, and pattern generation
- Rethinking innateness (Elman et al. 1996)

E.g. Handwritten character recognition through Hebbian competitive learning

- I.e. 10 neurons respond through scalar product with input
- From a random starting point, does not learn well: e.g. 7,8,9 confused

Synergy of Evolution and Learning



(d) Examples produced by an evolved pattern generator

(e) Weight vectors after prenatal training with evolved patterns

(f) Final weight vectors after additional competitive learning

Evolve prenatal pattern generators (Valsalam et al. 2007)

- Location, axes of Gaussians

Only a few units learn, but they establish a useful bias

- 7,8,9 will be separated in learning with the actual digits

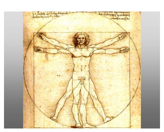
Could serve a maintenance function as well (Miikkulainen et al. 2005)

- Compensates for catastrophic forgetting

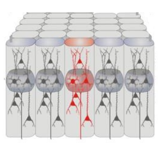
3.2. Generative Encodings

Desirable properties:

- Regularity
- Scalable



Symmetry

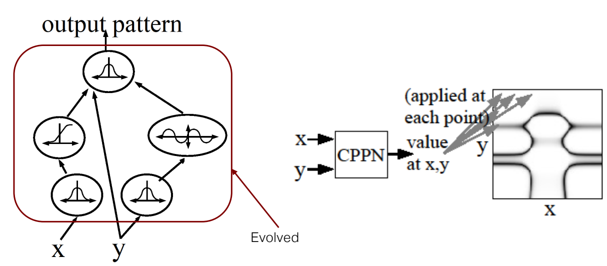


Repetition

Repetition with variation

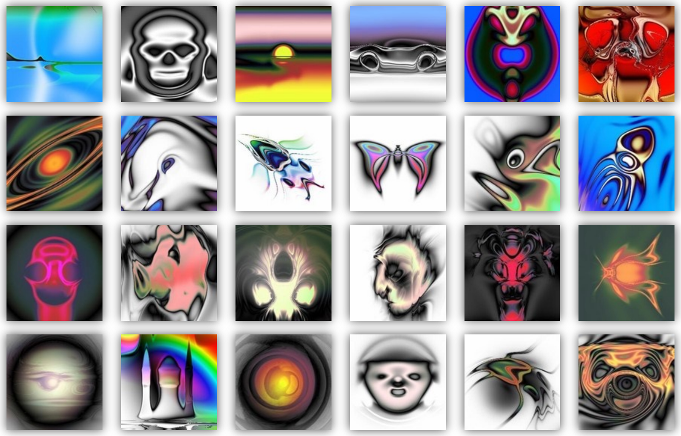
Compositional Pattern Producing Networks (CPPNs; Stanley 2007)

An artificial indirect encoding designed to abstract how embryos are encoded through DNA

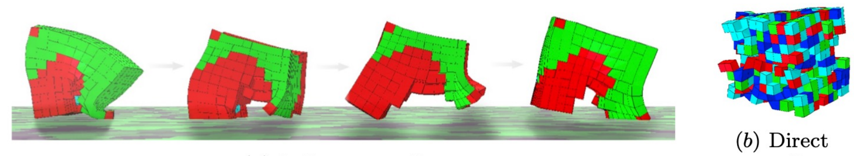


CPPN-Generated Patterns

www.picbreeder.org



From 2D Images to 3D Virtual Creatures

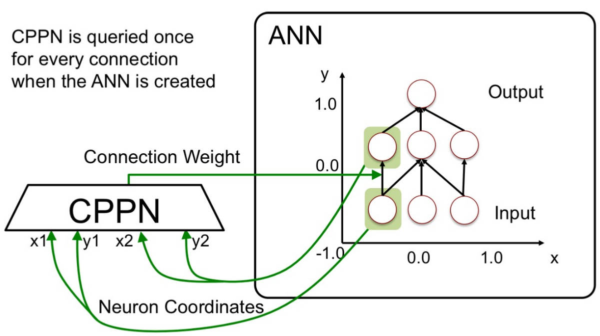


(a) Indirect encoding (b) Direct encoding

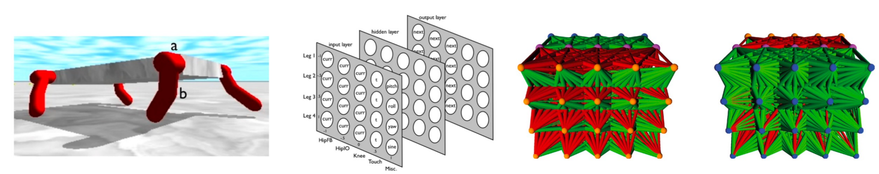
N. Cheney, R. MacCurdy, J. Clune, and H. Lipson. Unshackling evolution: evolving soft robots with multiple materials and a powerful generative en- coding. ACM SIGEVOlution, 7(1):11–23, 2014.

Encoding Brains Through CPPNs: HyperNEAT

(Stanley et al. 2009)



HyperNEAT-encoded Quadruped Locomotion

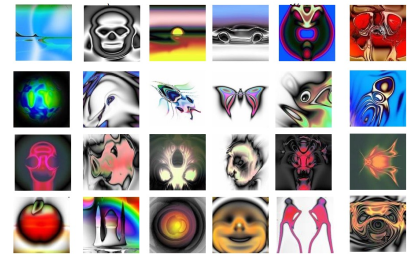


(a) (b) (c) (d)

J. Clune, K. O. Stanley, R. T. Pennock, and C. Ofria. On the performance of indirect encoding across the continuum of regularity. IEEE Transactions on Evolutionary Computation, 15(3):346–367, 2011

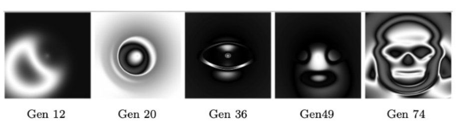
3.2 Diversity

3.2.1 Evolving for Novelty

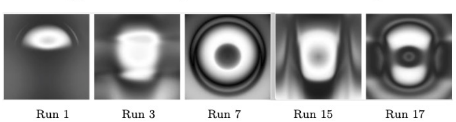


- Motivated by humans as fitness functions
- E.g. picbreeder.com, endlessforms.com[83]
  - CPPNs evolved; Human users select parents
- No specific goal
  - Interesting solutions preferred
  - Similar to biological evolution?

Novelty Search



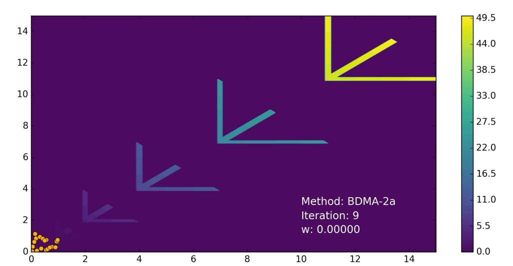
(*a*) Intermediate images in the evolution of the skull image



(*b*) Attempts to evolve the skull image directly

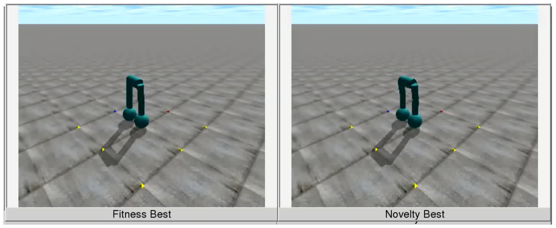
- Evolutionary algorithms maximize a performance objective
  - But sometimes hard to achieve it step-by-step
- Novelty search rewards candidates that are simply different[37,88]
  - Stepping stones for constructing complexity

Novelty Search Demo 1



- Illustration of stepping stones[49,50]
  - Nonzero fitness on “feet” only; stepwise increase
  - Top and right “toes” are stepping stones to next “foot”
  - Difficult for fitness based search; novelty can do it
- DEMO

Novelty Search Demo 2



- Fitness-based evolution is rigid
  - Requires gradual progress
- Novelty-based evolution is more innovative, natural[37,88]
  - Allows building on stepping stones
- How to guide novelty search towards useful solutions?
  - Quality Diversity methods[17,67]
- DEMO

3.2.2.Quality-Diversity

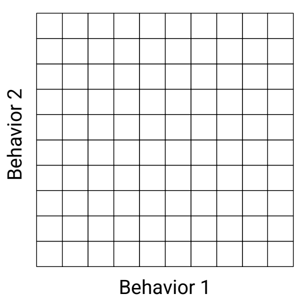
Goal: Find a set of different high-quality solutions for a given problem
Ex: Find fast walking gaits for a legged robot for every direction

Popular QD Method: Multi-dimensional Archive of Phenotypic Elites[1] (MAP-Elites)

Idea: Evolve an archive of different high-quality solutions (= elites)

[1]Mouret, J.-B. and J. Clune (2015). "Illuminating search spaces by mapping elites," In: arXiv preprint:1504.04909

Multi-dimensional Archive of Phenotypic Elites[1] (MAP-Elites)



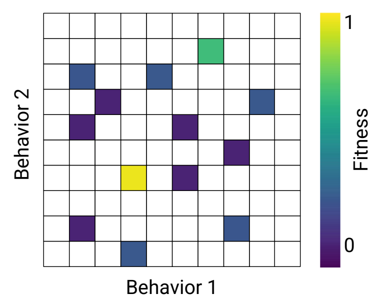
MAP-Elites[1] algorithm

Initialization
1. **Divide the behavior space into cells (= the Archive)**
2. Initialize with random solutions until n_{init} elites are found

Main loop
1. Pick two elites from the Archive
2. Apply crossover and mutation
3. Evaluate new solution
4. If new behavior or better fitness the solution becomes an elite

[1]Mouret, J.-B. and J. Clune (2015). "Illuminating search spaces by mapping elites," In: arXiv preprint:1504.04909

Multi-dimensional Archive of Phenotypic Elites[1] (MAP-Elites)



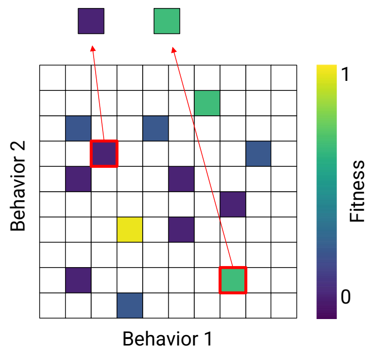
MAP-Elites[1] algorithm

Initialization
1. Divide the behavior space into cells (= the Archive)
2. **Initialize with random solutions until n_{init} elites are found**

Main loop
1. Pick two elites from the Archive
2. Apply crossover and mutation
3. Evaluate new solution
4. If new behavior or better fitness the solution becomes an elite

[1]Mouret, J.-B. and J. Clune (2015). "Illuminating search spaces by mapping elites," In: arXiv preprint:1504.04909

Multi-dimensional Archive of Phenotypic Elites[1] (MAP-Elites)



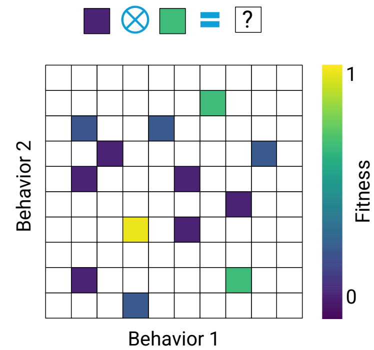
MAP-Elites[1] algorithm

Initialization
1. Divide the behavior space into cells (= the Archive)
2. Initialize with random solutions until n_{init} elites are found

Main loop
1. **Pick two elites from the Archive**
2. Apply crossover and mutation
3. Evaluate new solution
4. If new behavior or better fitness the solution becomes an elite

[1]Mouret, J.-B. and J. Clune (2015). "Illuminating search spaces by mapping elites," In: arXiv preprint:1504.04909

Multi-dimensional Archive of Phenotypic Elites[1] (MAP-Elites)



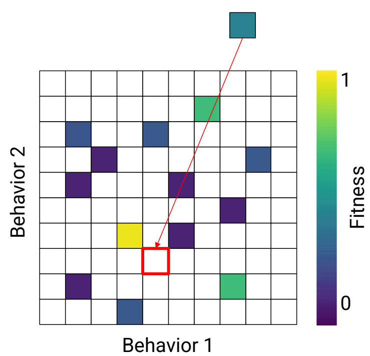
MAP-Elites[1] algorithm

Initialization
1. Divide the behavior space into cells (= the Archive)
2. Initialize with random solutions until n_{init} elites are found

Main loop
1. Pick two elites from the Archive
2. **Apply crossover and mutation**
3. Evaluate new solution
4. If new behavior or better fitness the solution becomes an elite

[1]Mouret, J.-B. and J. Clune (2015). "Illuminating search spaces by mapping elites," In: arXiv preprint:1504.04909

Multi-dimensional Archive of Phenotypic Elites[1] (MAP-Elites)



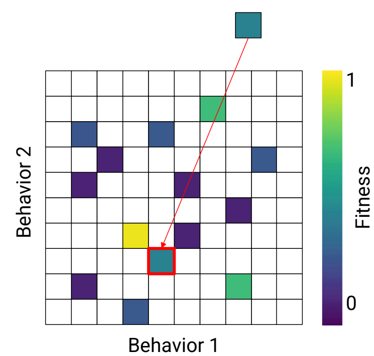
MAP-Elites[1] algorithm

Initialization
1. Divide the behavior space into cells (= the Archive)
2. Initialize with random solutions until n_{init} elites are found

Main loop
1. Pick two elites from the Archive
2. Apply crossover and mutation
3. **Evaluate new solution**
4. If new behavior or better fitness the solution becomes an elite

[1]Mouret, J.-B. and J. Clune (2015). "Illuminating search spaces by mapping elites," In: arXiv preprint:1504.04909

Multi-dimensional Archive of Phenotypic Elites[1] (MAP-Elites)



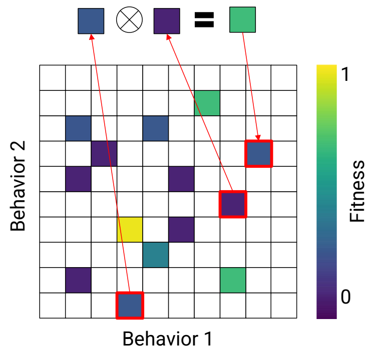
MAP-Elites[1] algorithm

Initialization
1. Divide the behavior space into cells (= the Archive)
2. Initialize with random solutions until n_{init} elites are found

Main loop
1. Pick two elites from the Archive
2. Apply crossover and mutation
3. Evaluate new solution
4. **If new behavior** or better fitness **the solution becomes an elite**

[1]Mouret, J.-B. and J. Clune (2015). "Illuminating search spaces by mapping elites," In: arXiv preprint:1504.04909

Multi-dimensional Archive of Phenotypic Elites[1] (MAP-Elites)



MAP-Elites[1] algorithm

Initialization

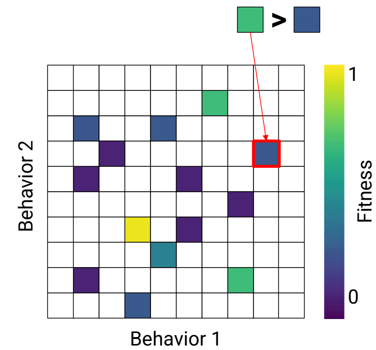
1. Divide the behavior space into cells (= the Archive)
2. Initialize with random solutions until n_{init} elites are found

Main loop

1. Pick two elites from the Archive
2. Apply crossover and mutation
3. Evaluate new solution
4. If new behavior or better fitness the solution becomes an elite

[1]Mouret, J.-B. and J. Clune (2015). "Illuminating search spaces by mapping elites," In: arXiv preprint:1504.04909

Multi-dimensional Archive of Phenotypic Elites[1] (MAP-Elites)



MAP-Elites[1] algorithm

Initialization

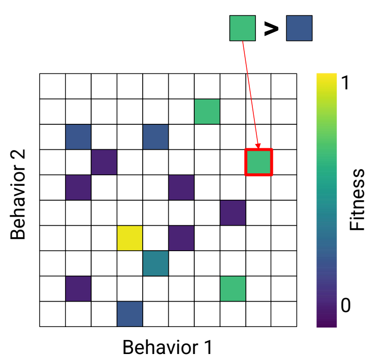
1. Divide the behavior space into cells (= the Archive)
2. Initialize with random solutions until n_{init} elites are found

Main loop

1. Pick two elites from the Archive
2. Apply crossover and mutation
3. Evaluate new solution
4. If new behavior or better fitness the solution becomes an elite

[1]Mouret, J.-B. and J. Clune (2015). "Illuminating search spaces by mapping elites," In: arXiv preprint:1504.04909

Multi-dimensional Archive of Phenotypic Elites[1] (MAP-Elites)



MAP-Elites[1] algorithm

Initialization

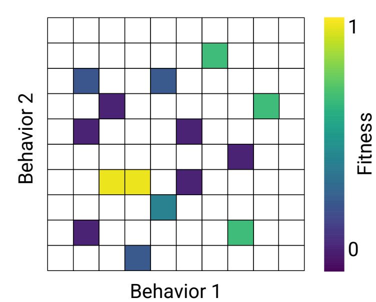
1. Divide the behavior space into cells (= the Archive)
2. Initialize with random solutions until n_{init} elites are found

Main loop

1. Pick two elites from the Archive
2. Apply crossover and mutation
3. Evaluate new solution
4. If new behavior or better fitness the solution becomes an elite

[1]Mouret, J.-B. and J. Clune (2015). "Illuminating search spaces by mapping elites," In: arXiv preprint:1504.04909

Multi-dimensional Archive of Phenotypic Elites[1] (MAP-Elites)



MAP-Elites[1] algorithm

Initialization

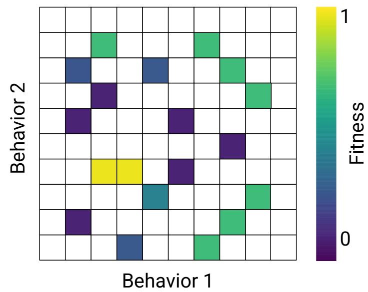
1. Divide the behavior space into cells (= the Archive)
2. Initialize with random solutions until n_{init} elites are found

Main loop

1. Pick two elites from the Archive
2. Apply crossover and mutation
3. Evaluate new solution
4. If new behavior or better fitness the solution becomes an elite

[1]Mouret, J.-B. and J. Clune (2015). "Illuminating search spaces by mapping elites," In: arXiv preprint:1504.04909

Multi-dimensional Archive of Phenotypic Elites[1] (MAP-Elites)



MAP-Elites[1] algorithm

Initialization

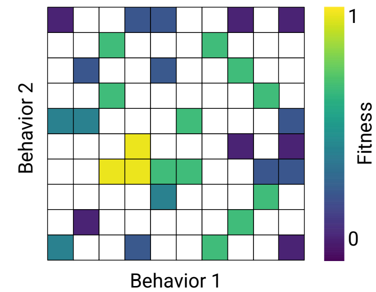
1. Divide the behavior space into cells (= the Archive)
2. Initialize with random solutions until n_{init} elites are found

Main loop

1. **Pick two elites from the Archive**
2. **Apply crossover and mutation**
3. **Evaluate new solution**
4. **If new behavior or better fitness the solution becomes an elite**

[1]Mouret, J.-B. and J. Clune (2015). "Illuminating search spaces by mapping elites," In: arXiv preprint:1504.04909

Multi-dimensional Archive of Phenotypic Elites[1] (MAP-Elites)



MAP-Elites[1] algorithm

Initialization

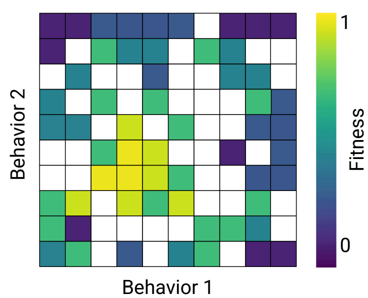
1. Divide the behavior space into cells (= the Archive)
2. Initialize with random solutions until n_{init} elites are found

Main loop

1. **Pick two elites from the Archive**
2. **Apply crossover and mutation**
3. **Evaluate new solution**
4. **If new behavior or better fitness the solution becomes an elite**

[1]Mouret, J.-B. and J. Clune (2015). "Illuminating search spaces by mapping elites," In: arXiv preprint:1504.04909

Multi-dimensional Archive of Phenotypic Elites[1] (MAP-Elites)



MAP-Elites[1] algorithm

Initialization

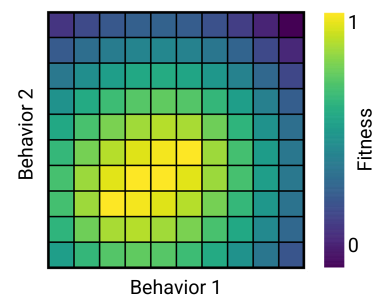
1. Divide the behavior space into cells (= the Archive)
2. Initialize with random solutions until n_{init} elites are found

Main loop

1. **Pick two elites from the Archive**
2. **Apply crossover and mutation**
3. **Evaluate new solution**
4. **If new behavior or better fitness the solution becomes an elite**

[1]Mouret, J.-B. and J. Clune (2015). "Illuminating search spaces by mapping elites," In: arXiv preprint:1504.04909

Multi-dimensional Archive of Phenotypic Elites[1] (MAP-Elites)



MAP-Elites[1] algorithm

Initialization

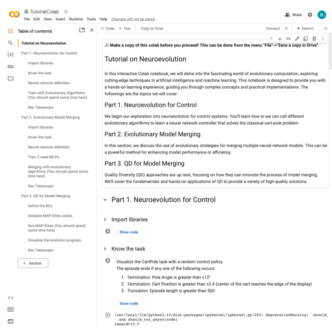
1. Divide the behavior space into cells (= the Archive)
2. Initialize with random solutions until n_{init} elites are found

Main loop

1. Pick two elites from the Archive
2. Apply crossover and mutation
3. Evaluate new solution
4. If new behavior or better fitness the solution becomes an elite

[1]Mouret, J.-B. and J. Clune (2015). "Illuminating search spaces by mapping elites," In: arXiv preprint:1504.04909

A Hands-on Exercise

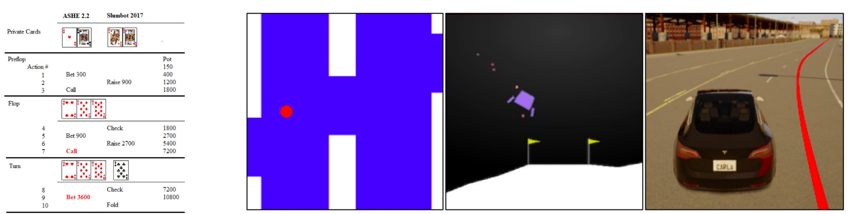


- Led by by Yujin Tang (remotely)
- https://colab.research.google.com/…
- Three parts:
  - Neuroevolution for Control
  - Evolutionary Model Merging
  - QD for Model Merging

4.1 Evolving Intelligent Agents

1. Control ®
   1. * Rocket, arm, satellite: Nonlinear, robust, innovative
   2. * Multilegged walking: symmetry
   3. * Flappyball: C+S, extrapolation
2. Strategy ®
   1. * Soccer: fractured
   2. * PacMan: multibehavior
   3. * Maze: deception

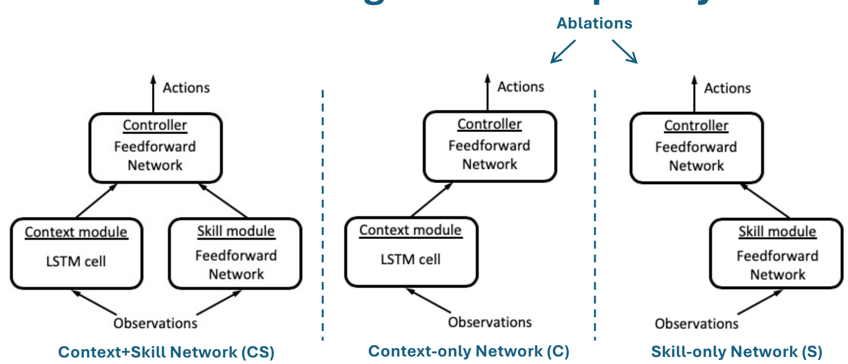
4.1.1.3 Coping with Unseen Situations



- The main challenge for evolved controllers: Extrapolation
- Need to generalize immediately to unseen situations
  - Games, robotics, process control, healthcare, finance

TEXAS
The University of Texas at Austin

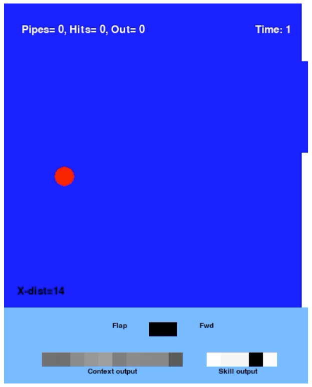
Solution: Modeling Context Explicitly



(Li et al 2018; Tutum et al. 2021)

TEXAS
The University of Texas at Austin

FlappyBall Domain

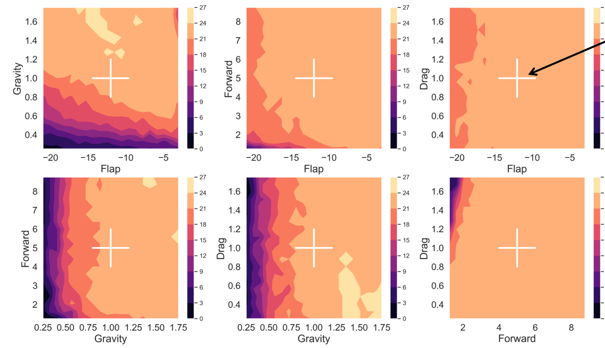


Demo

- Extension of Flappy Bird: FlapFwd, Drag
- **Inputs**: 6 numerical state values
  - Vertical position, distance to next pipe
  - Horizontal and vertical velocity
  - Height of the upper and lower pipe
- **Outputs**: select FlapUp, FlapFwd, glide
- **Objectives**:
  - Safety: Don't hit pipes, ceiling, ground
  - Performance: Fly fast
- **Task Variation:**
  - Strength of Gravity, Drag, FlapUp, FlapFwd

Illustration of Extrapolation

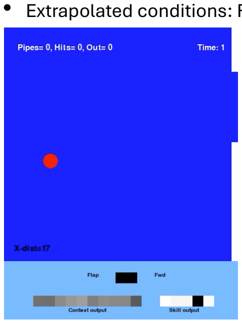
FB Performance; CS-S; Lighter is better; variation across pairs of parameters



- Training tasks distributed on the white cross
- Testing tasks distributed outside the cross: Require interpolation and significant extrapolation

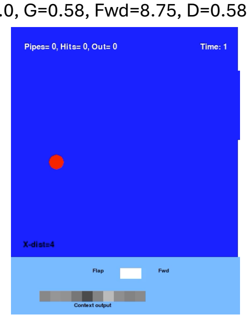
Example Behaviors in FlappyBall

- Extrapolated conditions: F=-7.0, G=0.58, Fwd=8.75, D=0.58



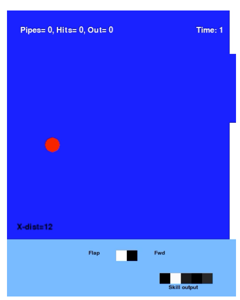
Context+Skill Network (CS)

Pipes = 21
Hits = 0



Context-only Network (C)

Pipes = 15
Hits = 6

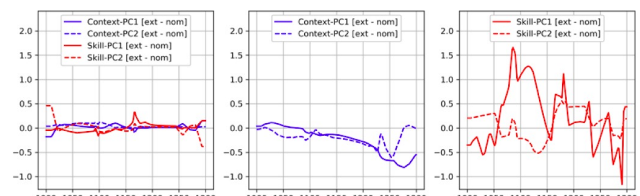


Skill-only Network (S)

Pipes = 16
Hits = 5

DEMO

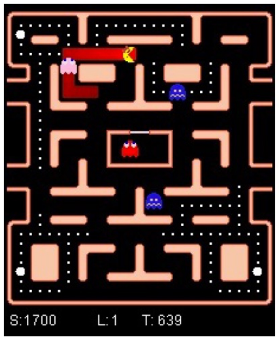
Modulation by Context



- Output of Context and Skill modules mapped to 2D with PCA
- Difference in an extrapolated task and the nominal task plotted
- Differences are smaller in CS than in C-only and S-only
  - Decision network needs to deal with less variance
  - Easier to generalize
  - CS evolves to make new tasks look more familiar
- Allows coping robustly in novel situations

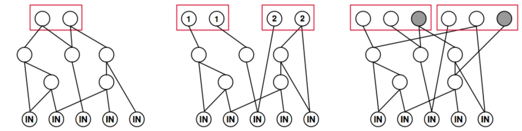
4.1.2.2

Challenge 1: Evolving Multimodal Behavior



- Agents perform many different tasks
  - E.g. eat pills, avoid ghosts, eat powerpills, eat ghosts
  - Sometimes clearly separate in time
  - Sometimes multiple tasks at once
- How can we evolve them into a single network?

MM-NEAT: Modular Multiobjective Approach



(a) Single-module Network (b) Multitask Network (c) Preference Neuron Network

- Evolution discovers modules and when to use them
  - Vs. human-designed division with multitasking
- Multiple modules with preference neurons[85,86]
  - Modules implement different behaviors
  - Preference neurons used to choose among them
  - Module-mutation adds new modules
- Evolved towards multiple objectives
  - Correspond to dimensions of game play
  - E.g. pills and ghosts in Ms. Pac-Man

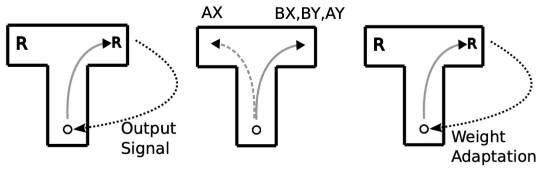
Human-Designed Task Division

- Multitask approach
  - One module for threat ghosts
  - Another module for edible ghosts
  - Works ok, but...
  - DEMO

Evolution-Discovered Task Division

- One module used 95% of the time
  - Eat pills, avoid ghosts, chase ghosts
  - Different behaviors with a common base
- A second module 5% of the time
  - Luring ghosts near a power pill
  - Escaping from tight spaces
- A different multimodal perspective
- Not as obvious, but more powerful
- DEMO

4.1.2.3 Search for Complex Behavior is Deceptive



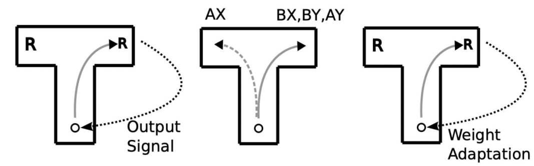
(a) Communication (b) Memory (c) Learning

(Lehman et al. 2014)

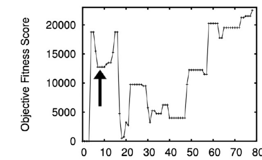
Cognitive tasks: Communication, Memory, Learning
- E.g. in Maze Navigation: turn left or right for reward

Need to discover multiple behaviors together:
- E.g. communication: what, when, send, receive, interpret
- Each one alone is not an advantage

Utilize Novelty Search Stepping Stones



(a) Communication (b) Memory (c) Learning



(d) Solution lineage

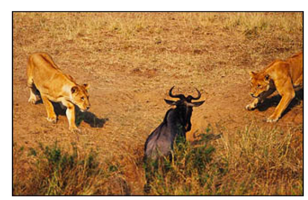
Possible with Novelty Search
- Each element is a stepping stone
- Fitness decreases as each is discovered
- Retained as local maxima and combined later

Abstraction of biological niching? Or weak selection and deep time?

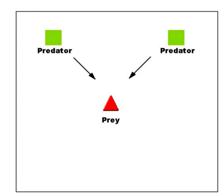
4.2 Collective Neuroevolution

1. Cooperative coevolution
2. Competive coevolution
3. Evolving neural cellular automata
4. Growing NNs with neural developmental programs

4.2.2.2 Coevolution of Behavior



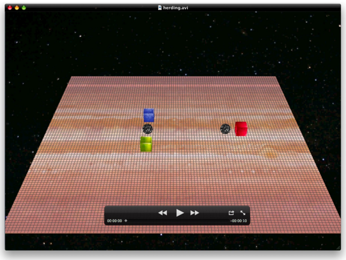
Natural predators and prey



Formalization of behavior

- Complex cooperation observed in pursuit and evasion
  - Motivated by biology, esp. hyenas vs. zebras (Kay Holekamp, MSU)
  - Largely innate, possible to see behaviors and their evolution
- Such behaviors evolve together, in coevolutionary environment
  - Simultaneous competitive and cooperative coevolution[73,75]

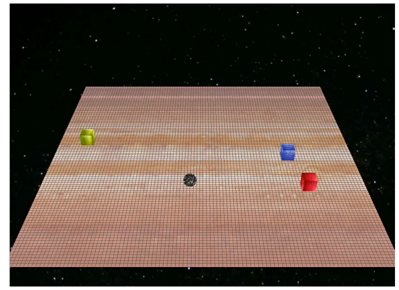
Experimental Setup



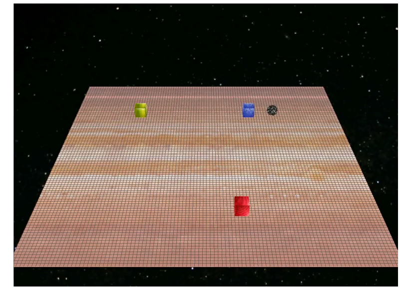
- Toroidal grid world
- Predators, prey move with same speed in 4 directions
- No direct communication between team members
  - Communication still possible through stigmergy
- Does a coevolutionary arms race result?

DEMO

Evolutionary Arms Race



50-75: Single predator catches the prey

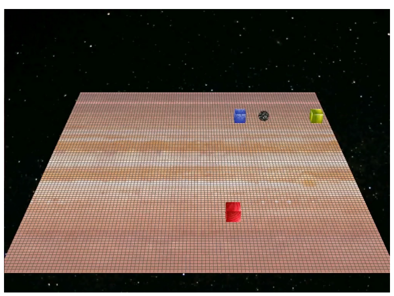


75-100: Prey evades by circling

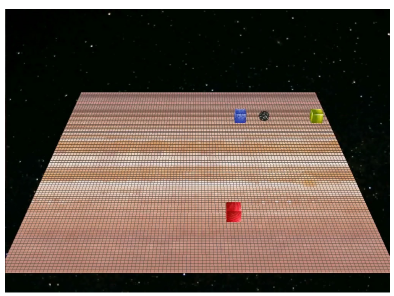
Predators and prey populations develop increasingly sophisticated behaviors (Rawal et al. 2010)

- Each improvement provides a challenge to the other population

Evolutionary Arms Race

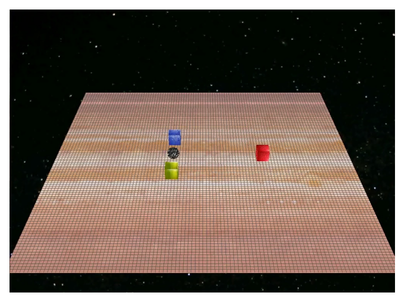


100-150: Two predators cooperate



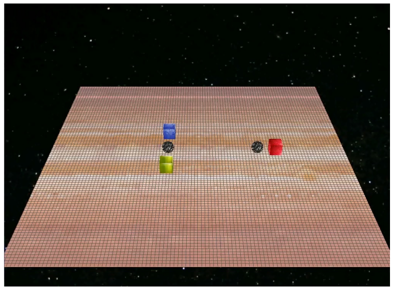
150-180: Prey baits and escapes

Evolutionary Arms Race

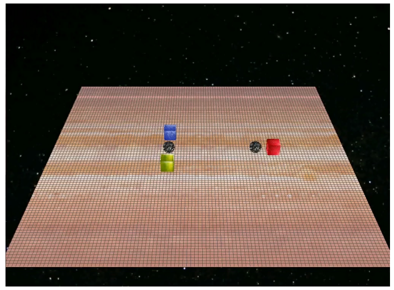


180-200: All predators cooperate

Evolutionary Arms Race

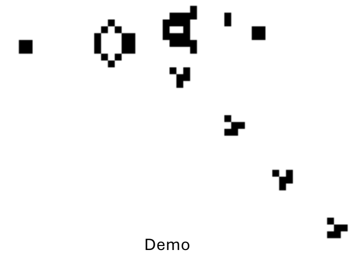


200-250: Predators herd two prey



250-300: Prey evade by scattering

4.2.3 Evolving Neural Cellular Automata

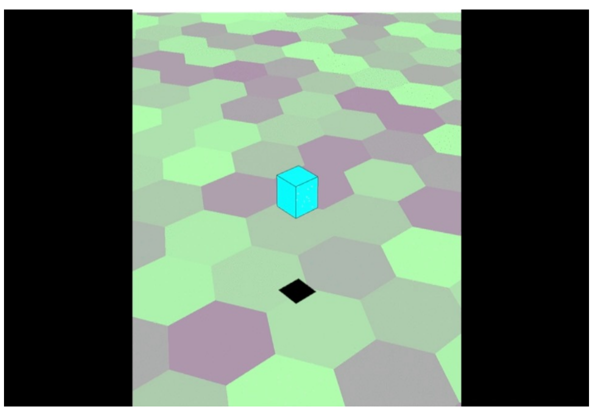


Demo

- Neural Cellular Automata (**Chua et al. 1988**)
- Differentiable NCA (**Mordvintsev et al. 2020**)

Regenerating Soft Robots through Neural Cellular Automata

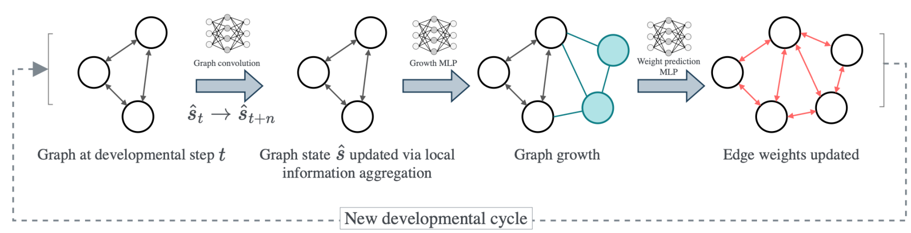
Horibe, Walker, Risi; EuroGP 2021



Demo

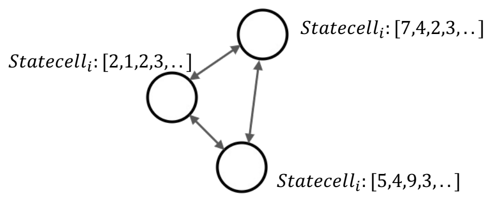
4.2.4 Growing NNs with neural developmental programs

Najarro, Sudhakaran and Risi, ALIFE 2023



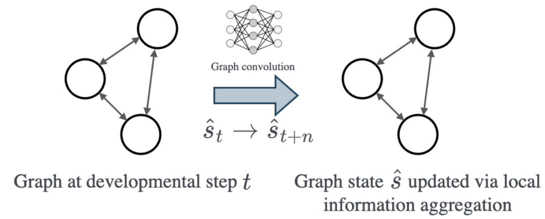
Self-Assembling ANNs through Neural Developmental Programs

Step 0: We start with an initial graph seed



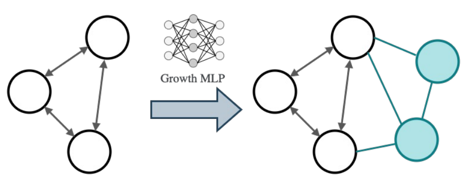
Self-Assembling ANNs through Neural Developmental Programs

Step 1: Update node states via message passing



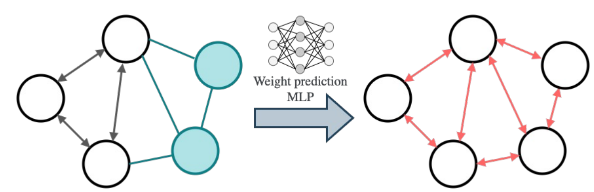
Self-Assembling ANNs through Neural Developmental Programs

Step 2: A neural network decides which nodes will grow

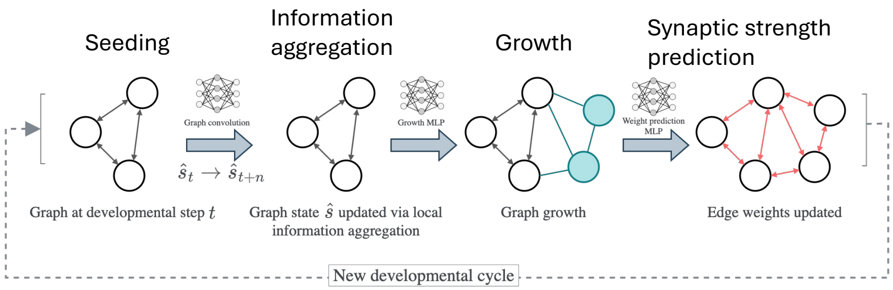


Self-Assembling ANNs through Neural Developmental Programs

Step 3: If network weighted: another NN determines edge weights

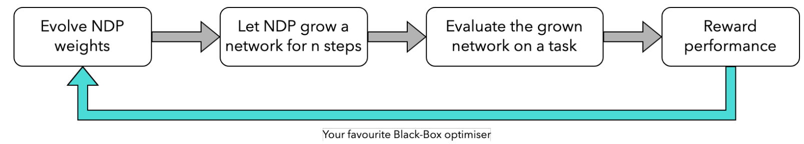


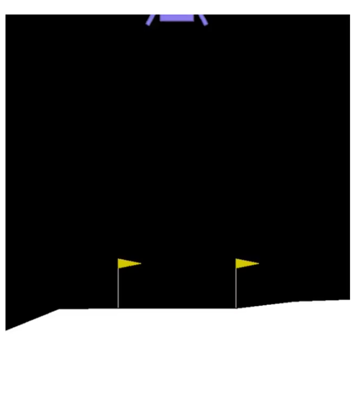
Self-Assembling ANNs through Neural Developmental Programs

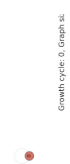


Self-Assembling ANNs through Neural Developmental Programs

Optimising the NDP to solve a task:







Demo

5.1 Neuroevolution and RL

Reinforcement Learning (RL)

- Advantages:
  - Handles sequential decision-making and dynamic environments.
  - Effective in domains with unknown or complex models
- Drawbacks:
  - Requires significant data and computational resources.
  - Training can be unstable and sensitive to hyperparameters.
  - Struggles with high-dimensional state and action spaces.

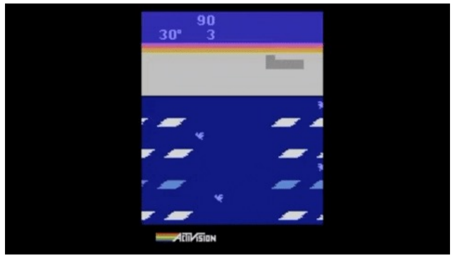
Neuroevolution

- Advantages:
  - More robust to local minima with population-based search.
  - Broader solution space exploration compared to RL.
  - Produces diverse policies.
- Drawbacks:
  - Generally slower than RL for real-time learning and adaptation.

NE vs RL

- Salimans et al. "Evolution Strategies as a Scalable Alternative to Reinforcement Learning", 2017.
- Such et al. "Deep Neuroevolution: Genetic Algorithms Are a Competitive Alternative for Training Deep Neural Networks for Reinforcement Learning", 2018.

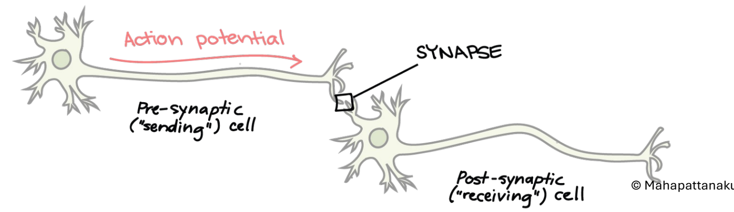
	DQN	ES	A3C	RS	GA	GA
Frames	200M	1B	1B	1B	1B	6B
Time	~7-10d	~ 1h	~ 4d	~ 1h or 4h	~ 1h or 4h	~ 6h or 24h
Forward Passes	450M	250M	250M	250M	250M	1.5B
Backward Passes	400M	0	250M	0	0	0
Operations	1.25B U	250M U	1B U	250M U	250M U	1.5B U
amidar	**978**	112	264	143	263	377
assault	4,280	1,674	**5,475**	649	714	814
asterix	4,359	1,440	**22,140**	1,197	1,850	2,255
asteroids	1,365	1,562	**4,475**	1,307	1,661	2,700
atlantis	279,987	**1,267,410**	911,091	26,371	76,273	129,167
enduro	**729**	95	-82	36	60	80
frostbite	797	370	191	1,164	**4,536**	**6,220**
gravitar	473	**805**	304	431	476	764
kangaroo	7,259	**11,200**	94	1,099	3,790	**11,254**
seaquest	**5,861**	1,390	2,355	503	798	850
skiing	-13,062	-15,443	-10,911	-7,679	†**-6,502**	†**-5,541**
venture	163	760	23	488	**969**	†**1,422**
zaxxon	5,363	6,380	**24,622**	2,538	6,180	7,864



Demo

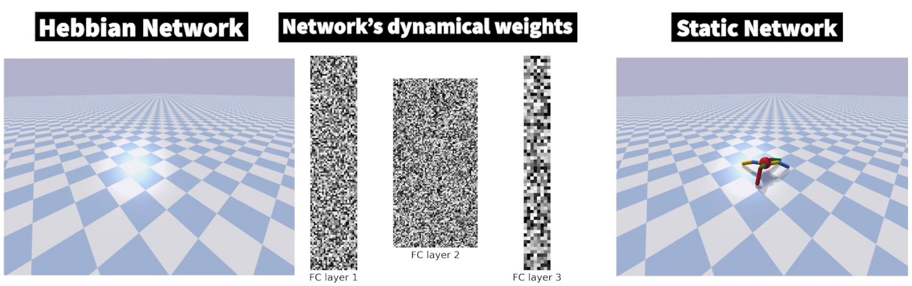
5.1.2. Meta-Learning through Hebbian Plasticity in Random Networks

Start network with random weights instead and only evolve local Hebbian learning rules ➔ can weights learn to self-organize?



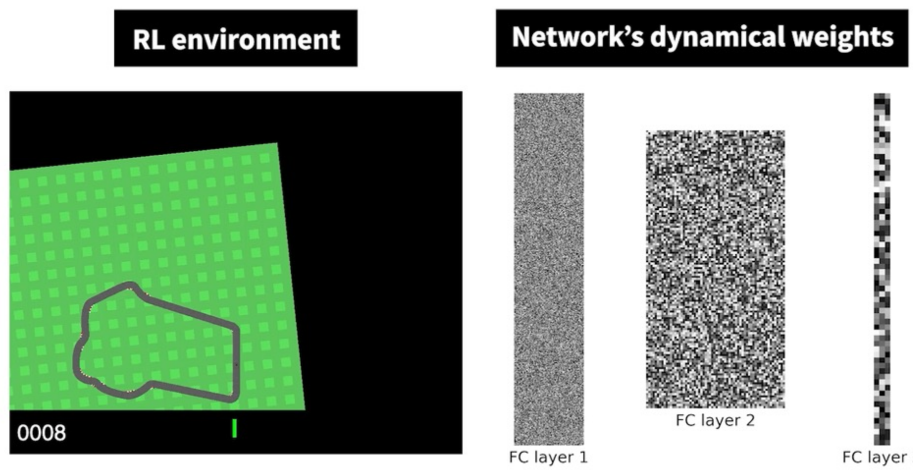
$$\Delta w_{ij} = \eta_w \cdot (A_w o_i o_j + B_w o_i + C_w o_j + D_w)$$

Najarro & Risi, NeurIPS 2020



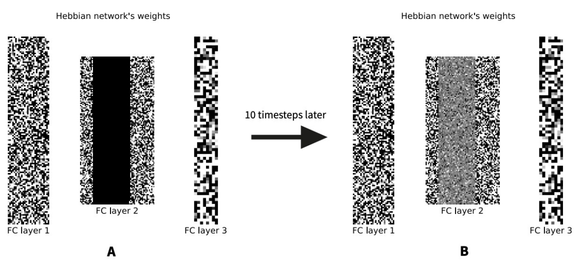
Seen during training

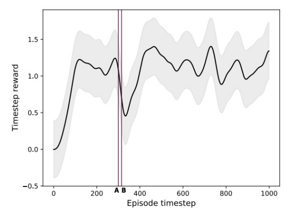
Demo



Demo

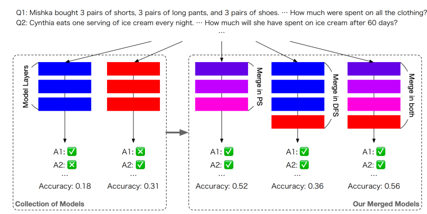
Resilience to Damage



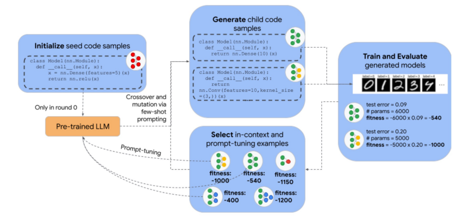


5.2 Neuroevolution and Generative AI

- Large Language Models have shown impressive performance across many domains
- Can be benefit from Neuroevolution methods in different ways



Evolutionary Model Merge (Sakana.ai, 2024)



EvoPrompting (Chen et al. 2023)

5.3 Neuroevolution Insights into Biology

1. Constrained evolution of behavior ®
   1. * EVC: Body/Brain
   2. * Botprize: resource limitations
2. Understanding evolutionary breakthroughs ®
   1. * Lions+hyenas: Evolution of intelligence
   2. * Mating/hunting: Origin of communication
3. Evolution of language ®
   1. * Signaling: common code (Werner), learning (Li), predicates (Cangelosi)
   2. * Language: Maybe cognition instead? Societies, roles, symbols

5.3.2.1 Emergence of Intelligence

Evolved Virtual Creatures
- Neuroevolution of intelligent behavior
- Useful e.g. for video games

Can such experiments lead to insights in biology?

Collaboration with Kay Holekamp's lab (MSU)
- Studying hyenas in Masai Mara since 1982

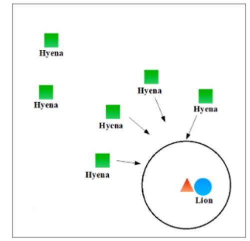
Example: Evolution of Intelligent Coordinated Behavior

Stealing a kill from lions

- Succeeds in an otherwise impossible task (sometimes)
- More sophisticated than other hyena behaviors
- Highly rewarding compared to normal hunting
- Largely genetically determined
- A breakthrough in evolution of intelligence?

DEMO

Simulation Setup



Lion at a kill, with an interaction circle around it

Ten hyenas chosen and placed randomly in the field

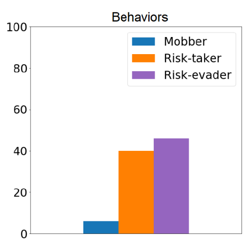
If 4 or more hyenas enter the circle simultaneously, they get the kill

- Otherwise they die

Does mobbing behavior evolve?

- What are the stepping stones for it?

Initial Behaviors



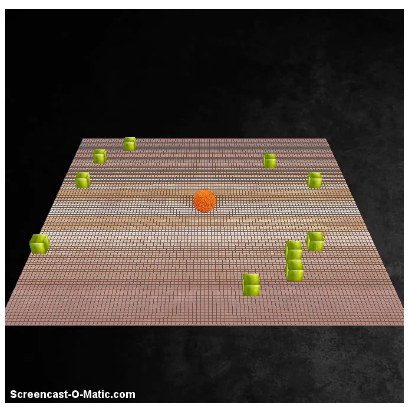
Risk evasion is common

- Never reach the circle; Medium fitness

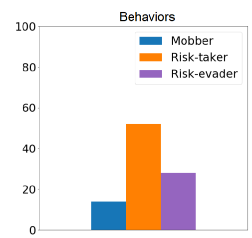
Risk taking is common

- Charge the circle; Frequent low fitness
- Occasional high fitness by accident

DEMO



Early Behaviors



Risk taking grows

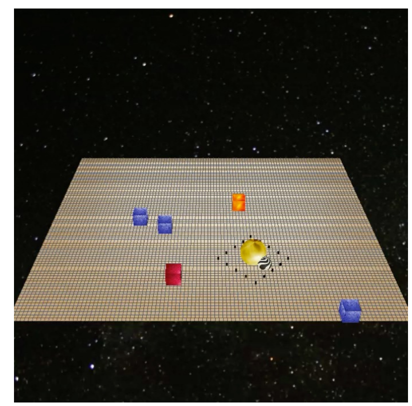
- As long as it is successful often enough

Risk evasion also persists

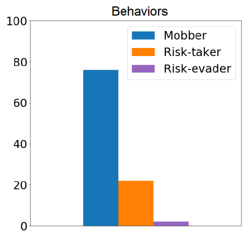
Evasion at the circle starts to emerge

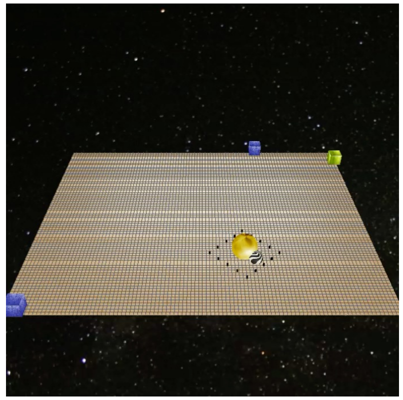
Is mostly detrimental, but an important stepping stone

DEMO



Later Behaviors



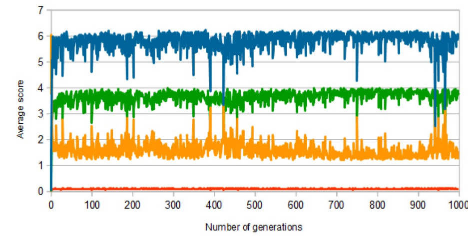


Mobbing emerges
- Not just coincidence of risk takers
- Hyenas wait until there's enough of them

Risk-evaders evolve into latecomers
Simple risk-taking and risk-evasion still exist

DEMO

These Behaviors Persist in Prolonged Evolution



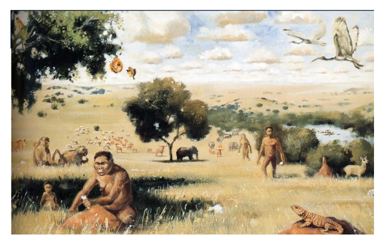
Risk taking and risk evasion never go away completely
- They serve a role in maintaining the mobbing behavior
- If mobbing starts to get lost, it can be reintroduced

Insight into Real-life Behaviors

These behaviors are observed in real-life hyenas as well

A computational explanation of why they are there:
- Stepping stones in discovery
- Safeguards in maintaining

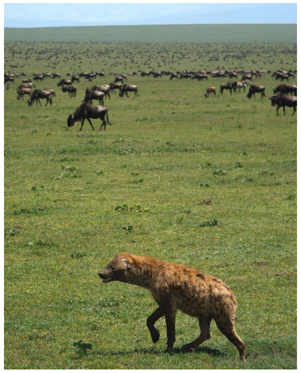
5.3.2 Challenge: Evolution of Language



Signaling is possible to evolve in ecological simulations
Structured language is much harder
Perhaps language evolved not from signaling, but cognition
- Complex social structure, with modifiable roles
- Language structure can reuse the same conceptual structures

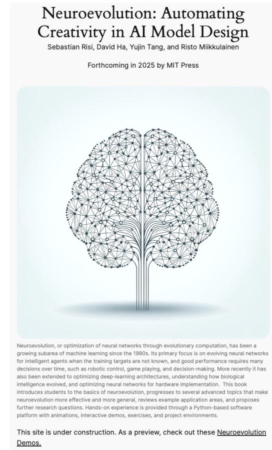
Enough compute, complex simulations to study now?

6. Conclusion: Constructing Alife Systems



- Believable, complex behavior in embedded environments
  - Open-ended "arms race" [72]
- Similar to self-play e.g. in AlphaGo Zero
  - Complexity beyond human ability to design it
- If we can build open-ended environments, we should be able to build more complex solutions
  - Co-evolve environments and behaviors? (e.g. POET,[100] EUREQA[80])
  - Evolution of memory, learning, language
- Challenge: Establish major transitions [57]

Need a Deeper Dive? Check out the Neuroevolution Book!



- Forthcoming in 2025
- Broad overview, current directions
- Demos
- Software platform: Exercises, projects
- https://neuroevolutionbook.com