Recent Advances in Population-Based Search

Quality Diversity, Open-Ended Algorithms, and Indirect Encodings


Jeff Clune


Joel Lehman


Ken Stanley

UBER AI Labs

Goal

- Share ideas that are
  - exciting
    - powerful: enable us to solve previously unsolved problems
    - insightful
    - true path
  - not well-known in ML, but useful in ML
    - developed outside traditional ML community
      - population-based methods
    - but broadly applicable
      - non-population based methods (e.g. RL, deep learning)
      - beyond neural networks
        - decision trees, program synthesis, etc.

Goal

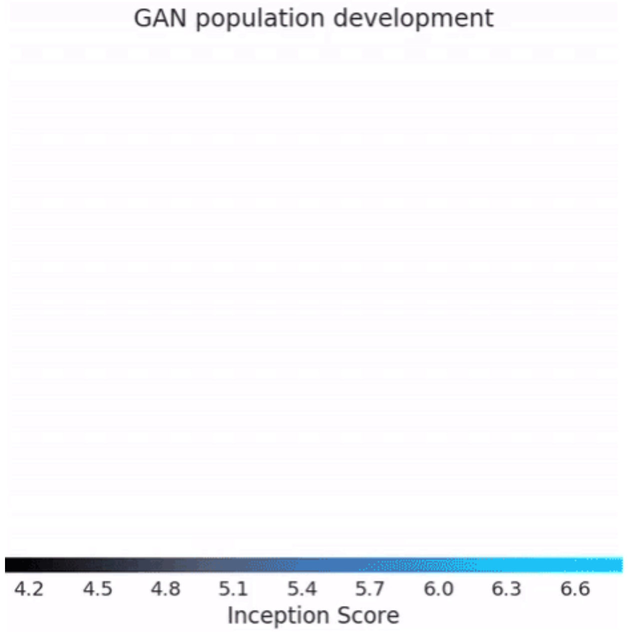
- introduce
  - new methods
  - new types of problems
    - including two grand challenges

Topics Covered & Schedule

- Novelty Search
- Quality Diversity
- Q&A (5 minutes)
- Open-Ended Search
- Indirect Encoding
- Looking Forward & Conclusions
- Q&A

Population-based Search

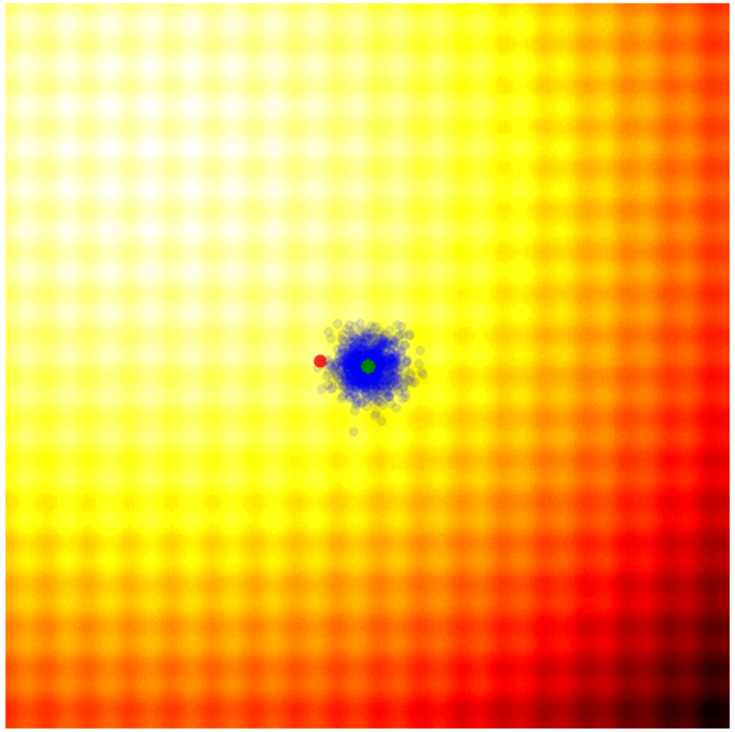
- Main idea: Maintain a *population* of candidate solutions



From: Deepmind Blog post on PBT

Population-based Search

- Canonical example:
  Vanilla Genetic Algorithm
  - Randomly initialize all members of population
  - Iteratively:
    - Evaluate population
    - Cull population
    - Make noisy copies
- Not a convincing case for benefits of a population
  - Convergent
  - One BBO among many



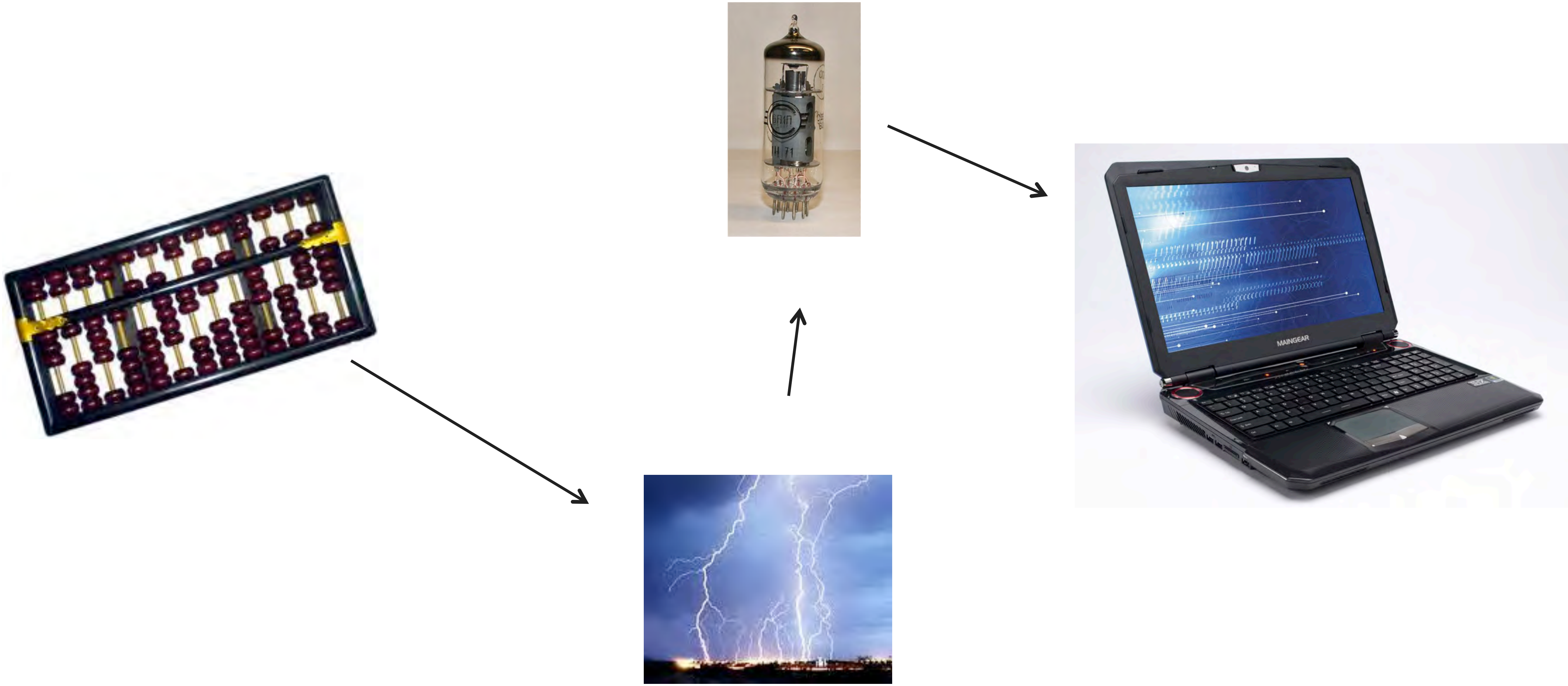
from David Ha

Diversity-centric Search

- Encouraging diversity as a central drive
- Novelty search (Lehman and Stanley 2008)
  - What would a search process driven only by diversity look like?
- Hypothesis: Diversity-centric search might be necessary to scale to our most ambitious ML objectives
  - Why?

Objectives and Objective Functions

- Objective functions are ubiquitous in ML
  - Measure of quality of a solution
  - Implicitly defines an objective to reach (by optimizing OF)
- The issue of local optima
  - Sometimes objective functions are smooth and easy to optimize
  - Sometimes optimization is more difficult because of thorny local optima
- Would our problems be solved if we simply created more powerful optimization algorithms?

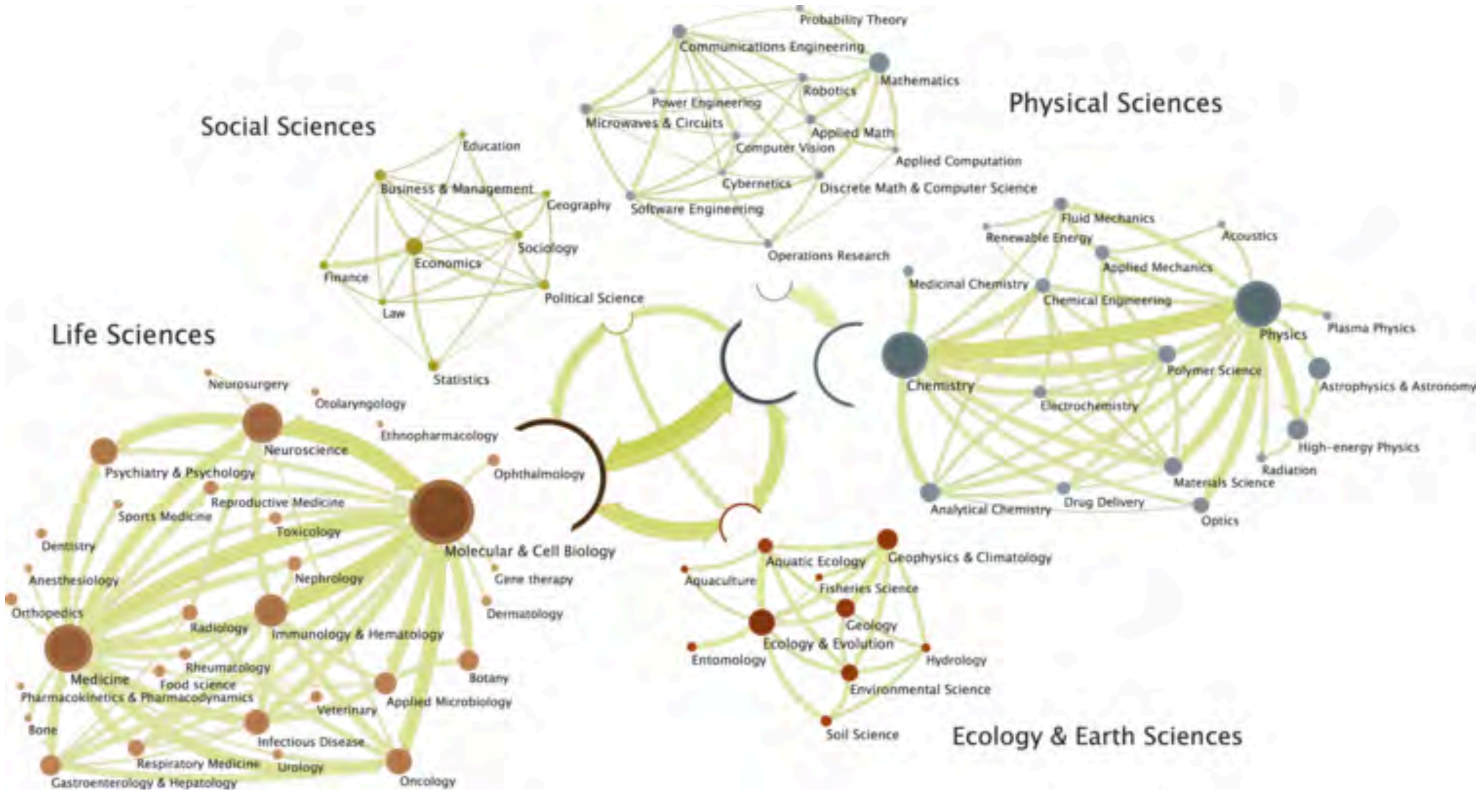

MAINGEAR

Deception

- The problem of **deception**: When aimed at ambitious objectives, the objective function often becomes a false compass
- Stepping stones to objective often seemingly unrelated to objective
  - From abacuses to laptops [electricity, vacuum tubes]
  - From prokaryotes to humans [multicellularity, development, neurons]
  - From random init to highly-intelligent robotic control policies [?]

The Problem with Ambitious Objectives

- Hopeful assumption: Improved performance will lead to greater improvements, all the way to success
- Doesn’t always work (local optima), which motivates:
  - Curriculum learning (Bengio et al. 2009)
  - Reward shaping/engineering (Ng et al. 1999)
  - Intrinsic motivation (Oudeyer and Kaplan 2007, Schmidhuber 1991)
  - Optimal reward functions (Singh et al. 2010)
- Overarching issue:
  Stepping stones to success don’t always resemble success



Social Sciences
Physical Sciences
Life Sciences
Ecology & Earth Sciences
Probability Theory
Communications Engineering
Mathematics
Robotics
Power Engineering
Microwaves & Circuits
Applied Math
Computer Vision
Applied Computation
Cybernetics
Discrete Math & Computer Science
Software Engineering
Operations Research
Education
Business & Management
Geography
Sociology
Economics
Finance
Law
Political Science
Statistics
Fluid Mechanics
Renewable Energy
Acoustics
Applied Mechanics
Medicinal Chemistry
Chemical Engineering
Physics
Plasma Physics
Polymer Science
Chemistry
Astrophysics & Astronomy
Electrochemistry
High-energy Physics
Radiation
Materials Science
Analytical Chemistry
Drug Delivery
Optics
Neurosurgery
Otolaryngology
Ethnopharmacology
Neuroscience
Psychiatry & Psychology
Ophthalmology
Reproductive Medicine
Sports Medicine
Toxicology
Molecular & Cell Biology
Dentistry
Nephrology
Gene therapy
Anesthesiology
Dermatology
Orthopedics
Radiology
Immunology & Hematology
Medicine
Rheumatology
Botany
Food science
Pharmacokinetics & Pharmacodynamics
Applied Microbiology
Veterinary
Bone
Infectious Disease
Respiratory Medicine
Urology
Gastroenterology & Hepatology
Oncology
Aquatic Ecology
Geophysics & Climatology
Aquaculture
Fisheries Science
Geology
Ecology & Evolution
Entomology
Hydrology
Environmental Science
Soil Science

Towards more creative search

- Radical idea:
  Can search that is ignorant of its intended objective sometimes ***outperform*** search that is aimed directly at its objective?
  - Can pursuing an ambitious objective undermine attaining it?
- What could instantiate a more open-ended search?
  - Creative, divergent forces?

Novelty Search

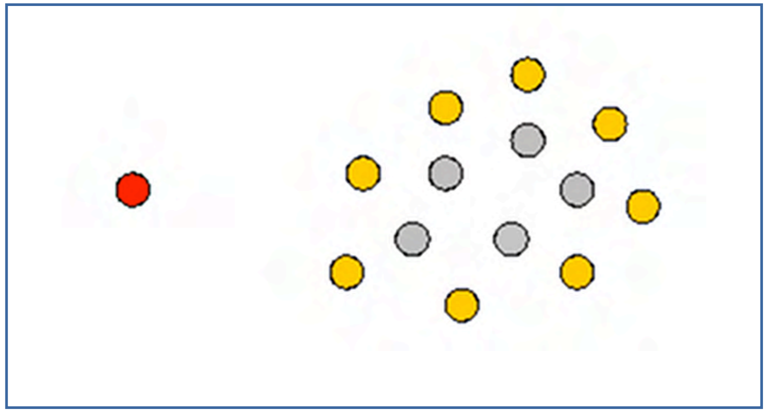
- Guiding search *only by novelty*
- Objective-driven heuristic: What improves performance locally is a stepping stone towards great performance
- Novelty-driven heuristic: What is novel may lead to further novelties

Novelty Search Algorithm

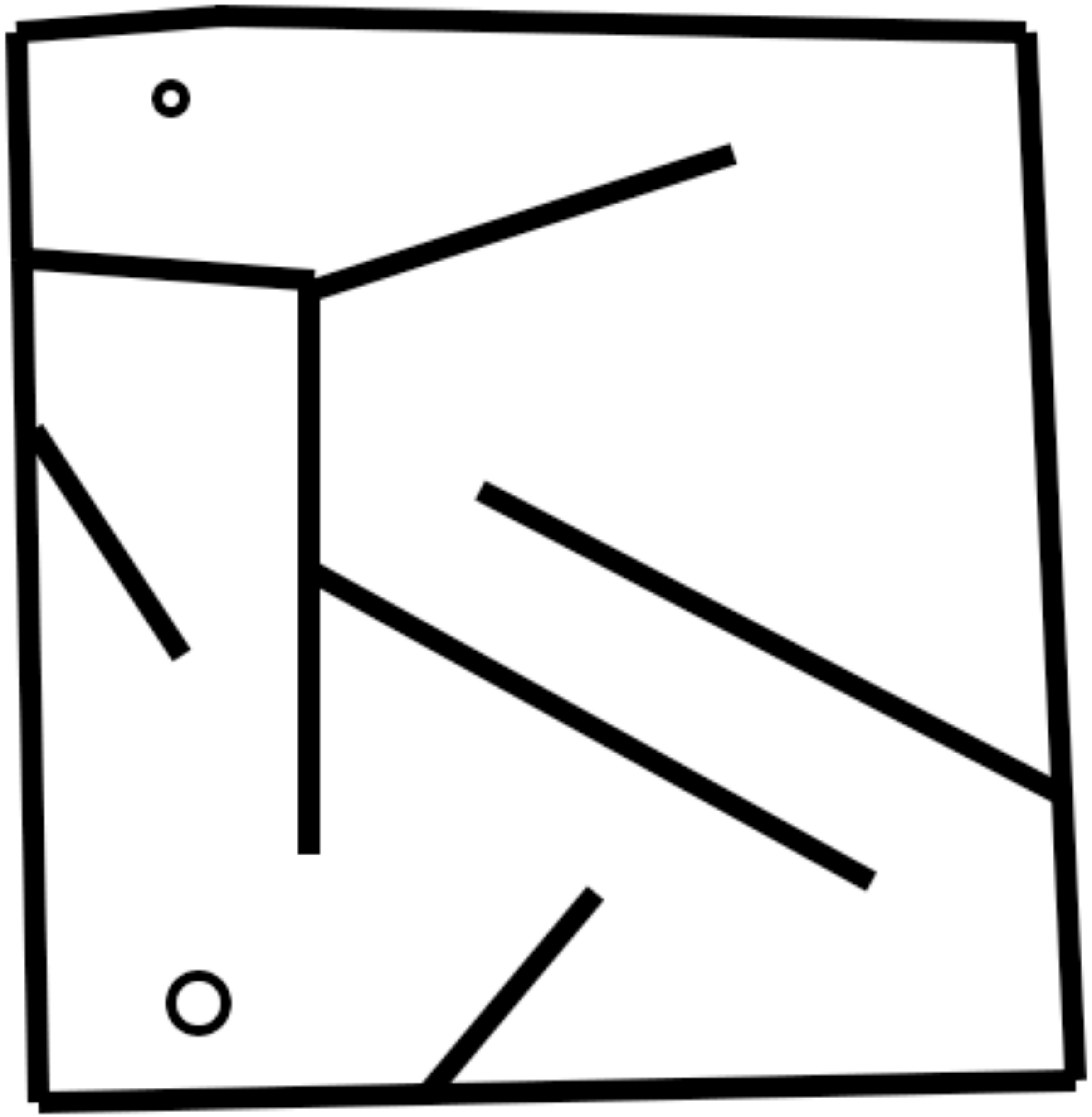
- Take a population-based search algorithm
  - Replace standard goal-based objective function with measure of *behavioral novelty*
    - Measured relative to *current population* and archive of previously-novel
- Over generations, search spreads out over the behavior space

$$\rho(x) = \frac{1}{k} \sum_{i=0}^{k} \mathrm{dist}(x, \mu_i)$$

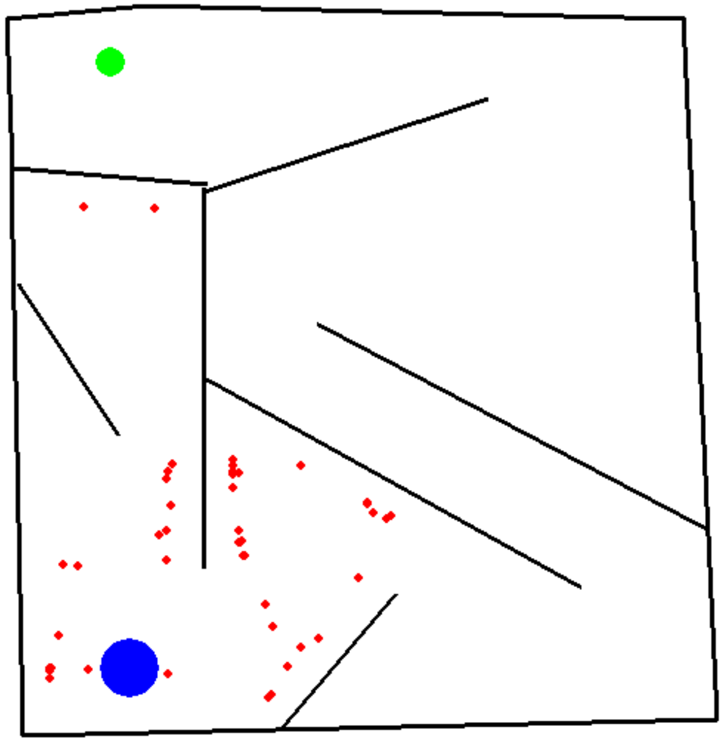
k-Nearest Neighbors
distance

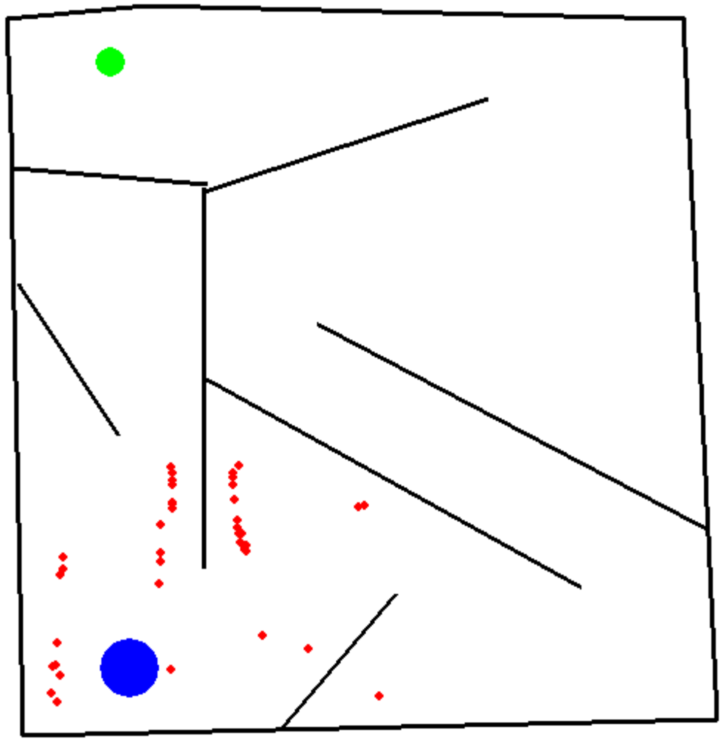


Behavior space



Visualization in Maze Navigation

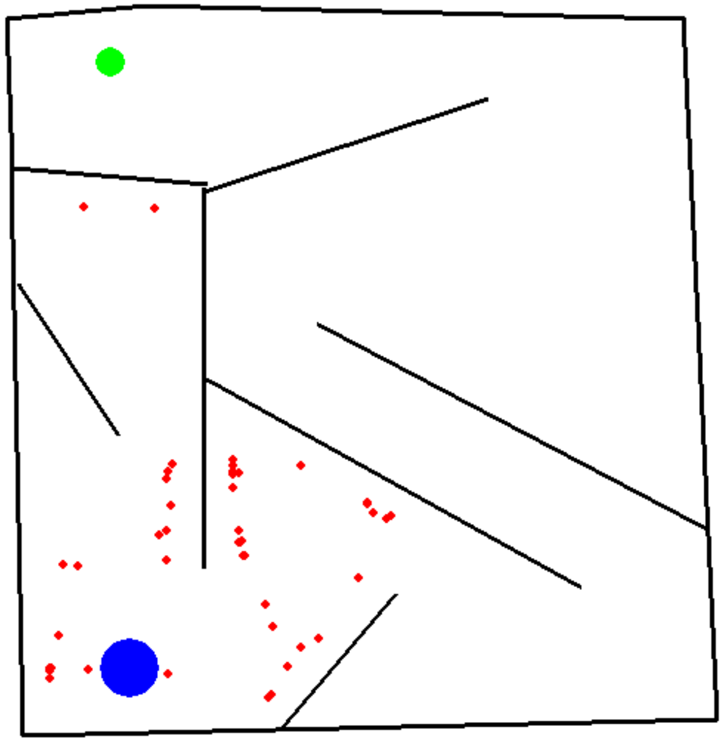




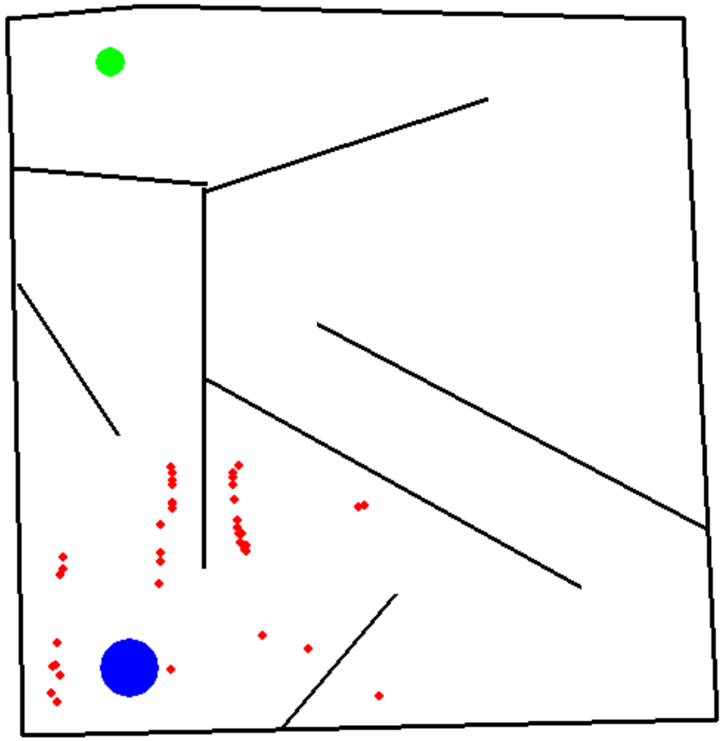
(Lehman and Stanley 2008)

Visualization in Maze Navigation

Novelty

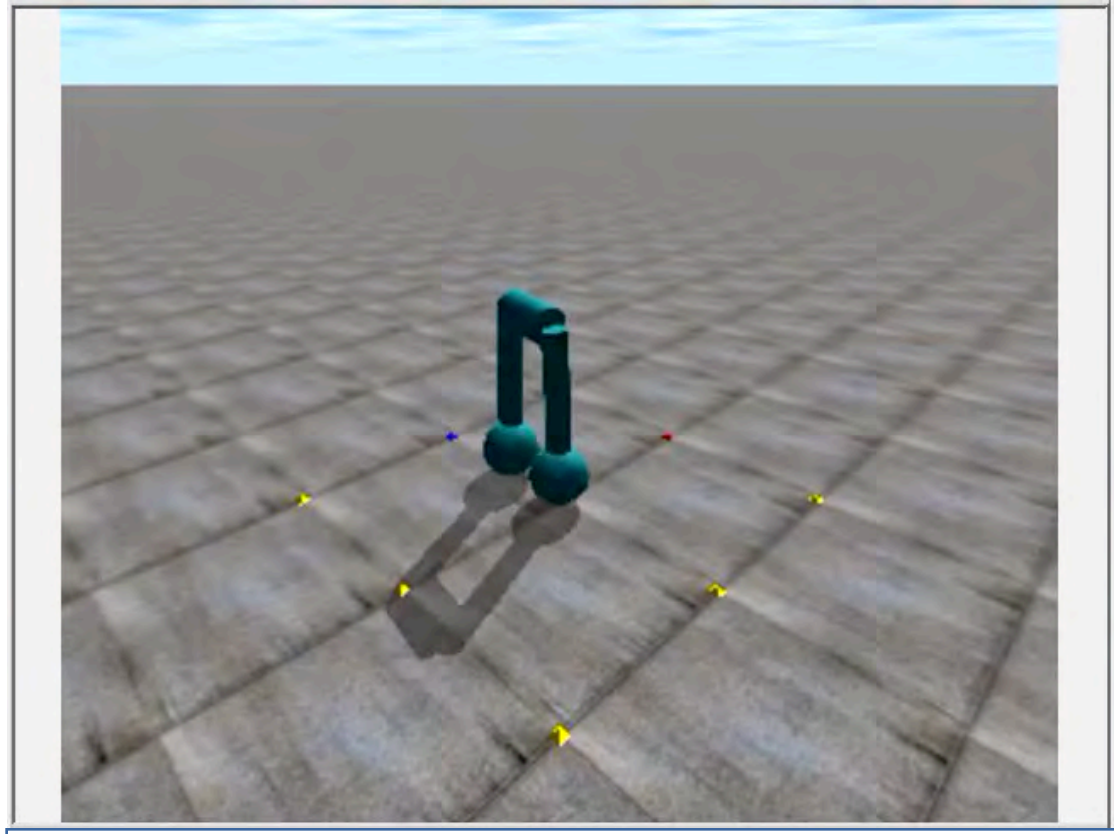


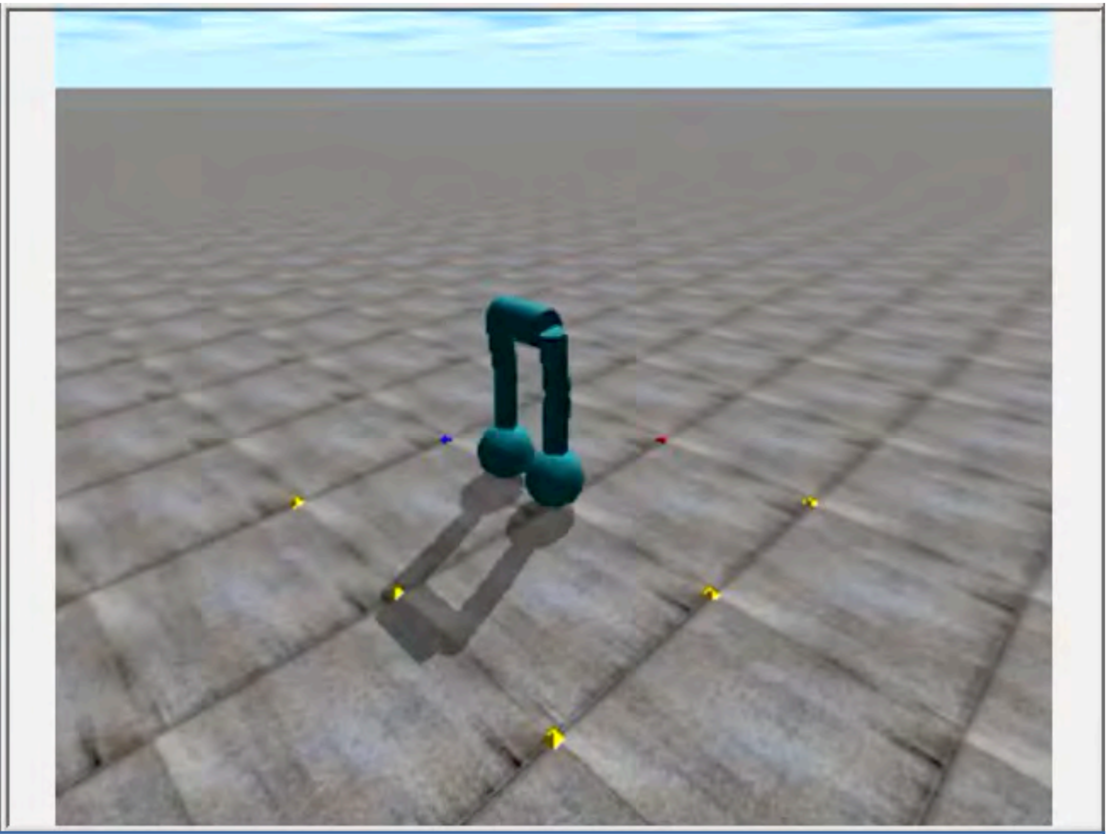
Objective



(Lehman and Stanley 2008)

Biped Locomotion





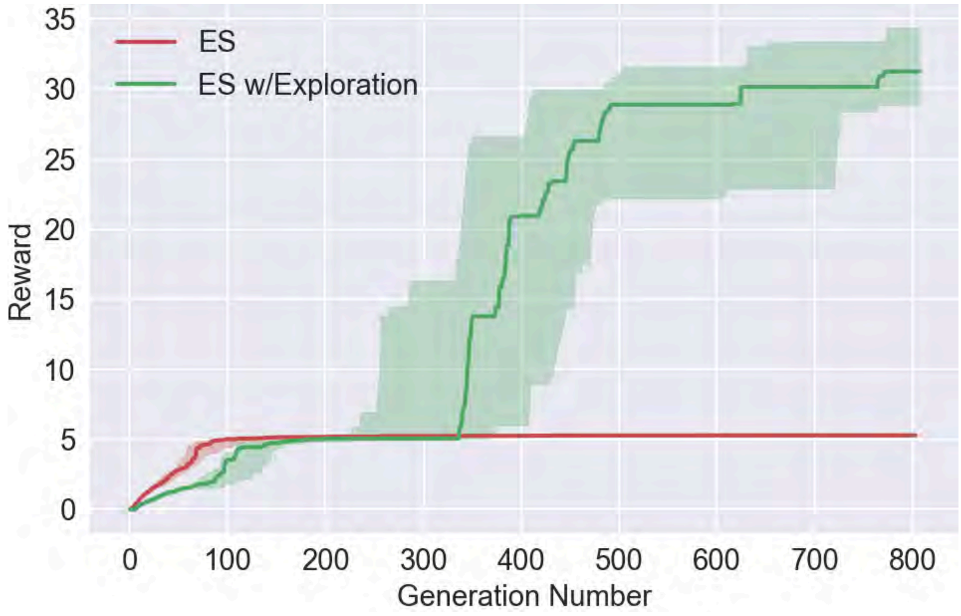
Objective	Novelty

(Lehman and Stanley 2012)

Works in Deep RL context too

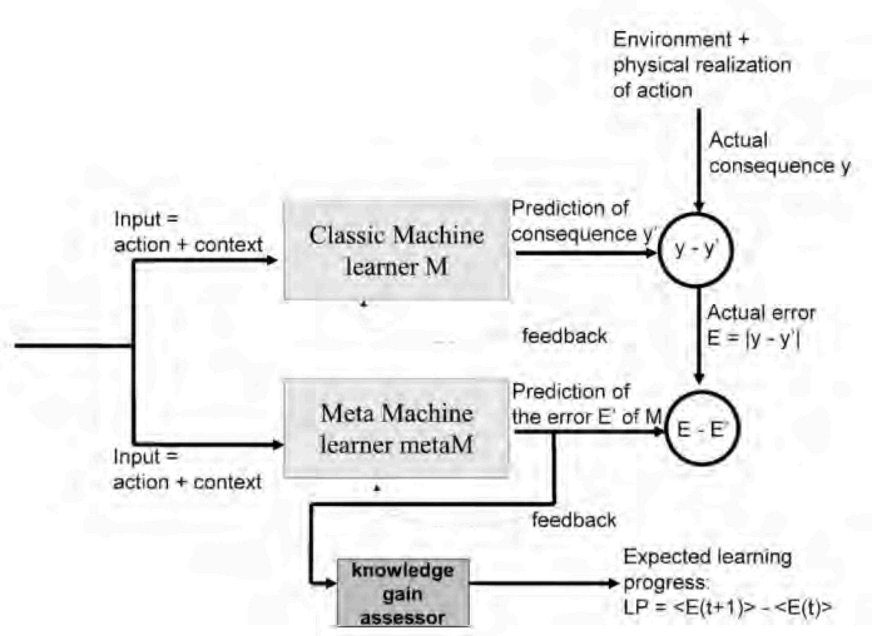
- As an extension of OpenAI's ES (Conti et al. 2018)
- As an extension of Uber's Deep GA (Such et al. 2017)

(Conti et al. 2018)



Related Work

- See also:
  Autonomous mental development / intrinsic motivation / curiosity (Oudeyer and Kaplan 2007, Schmidhuber 1991)

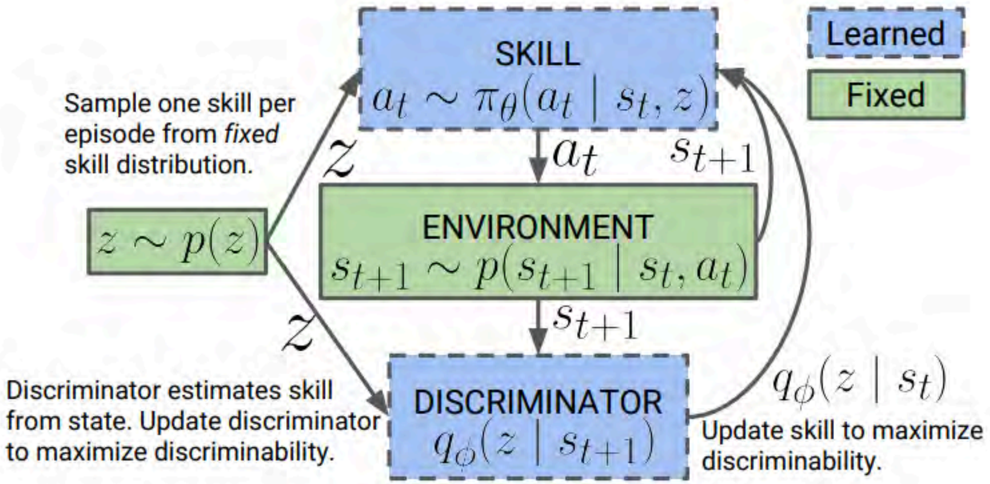


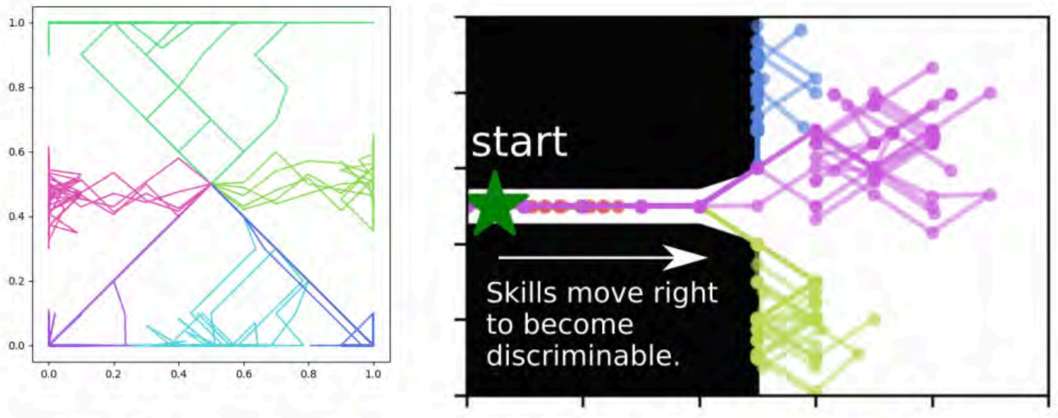
From: (Oudeyer et al. 2007)

Related Ideas in Deep RL

- DIAYN (Eysenbach et al. 2018)
- Curiosity-driven exploration (Pathak et al. 2017)
- Skew-fit (Pong et al. 2019)
- Hindsight Experience Replay (Andrychowicz et al. 2017)
- Unsupervised Meta-learning (Gupta et al. 2018)

Diversity is All You Need: Learning Diverse Skills without a Reward Function

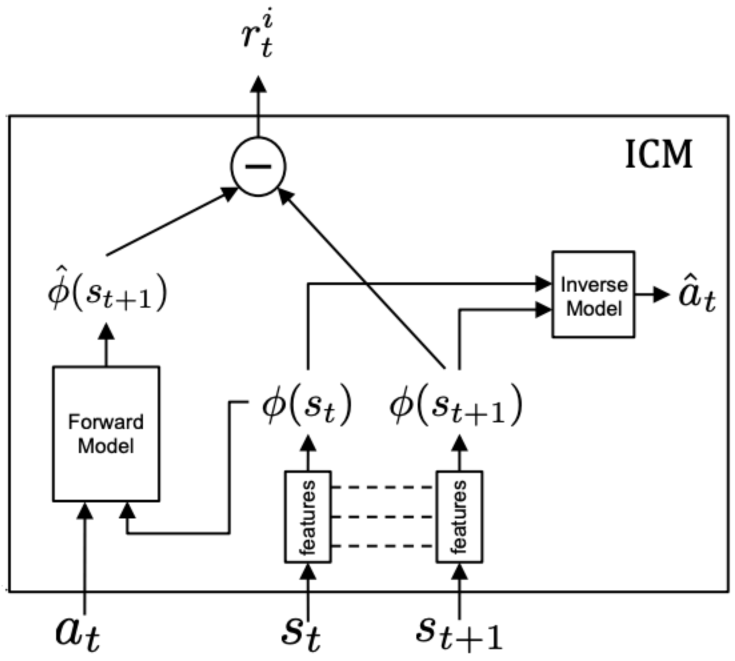




(a) 2D Navigation (b) Overlapping Skills

(Eysenbach et al. 2018)

Curiosity-driven Exploration by Self-Supervised Prediction



(a) learn to explore on Level-1 (b) explore faster on Level-2

(Pathak et al. 2017)

Novelty Search Conclusions

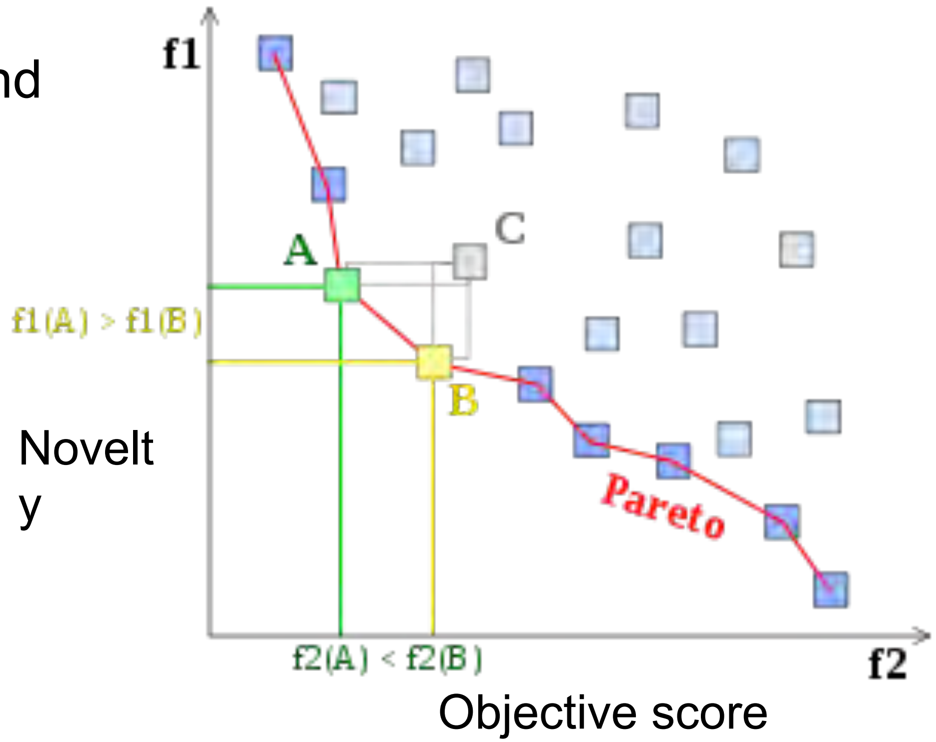
- Pressure towards creative divergence ***alone*** can sometimes outperform directly seeking the objective
- But what about the pressure to achieve (also a key force in biological and technological evolution)?

Combining Novelty and Achievement (Mouret and Doncieux 2012)

- While raw novelty can work, natural to merge novelty pressure with pressure to achieve
  - Many paradigms: Weighted average of objective + novelty; objective until stuck, then switch to novelty; etc.
- Effective in practice: Population-based multi-objective optimization
  (Fonseca et al. 1995)
  - Simultaneously explore all trade-offs between objectives

Population-based Multi-objective Optimization

- Popular algorithms include NSGA-II (Deb et al. 2002)
- Main idea: Maintain *pareto front* of non-dominated solutions
  - A>B only if
    - objective_score(A) > objective_score(B) and
    - novelty(A) > novelty(B)
- Another interesting possibility enabled by maintaining a population



Diversity + Performance as Equals

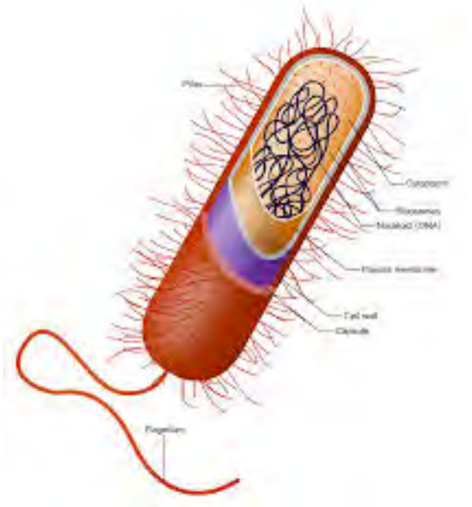
- Problems with combining novelty and *global competition* objective
  - Does not address the fundamental problem of deception
  - Embodies paradigm of diversity *in service of* progress
- What about an algorithm with equal priority to diversity and performance?
  - To optimize towards the best version of each possible solution niche?

Quality Diversity (Pugh et al. 2016)

- Different kind of search process:
  Find the best possible example of each achievable behavior
- Build a *repertoire* of different ways to solve a problem
  - Highlights a wide range of possible designs that a designer can choose from
  - Can enable a robot to adapt to new circumstances
  - Can circumvent deception by creating an implicit curriculum

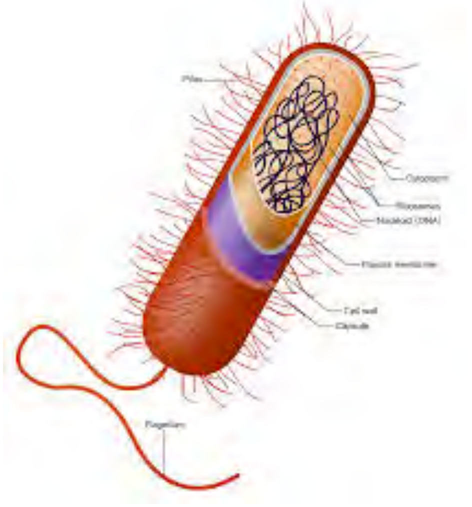
Quality Diversity

- Sometimes objective performance not the most important factor
  - Illuminate the space of diverse possible solutions
  - Diversity in *how* a problem is solved sometimes more important/ interesting than gaining only the single-most efficient solution



Quality Diversity

- Sometimes objective performance not the most important factor
  - Illuminate the space of diverse possible solutions
  - Diversity in *how* a problem is solved sometimes more important/ interesting than gaining only the single-most efficient solution

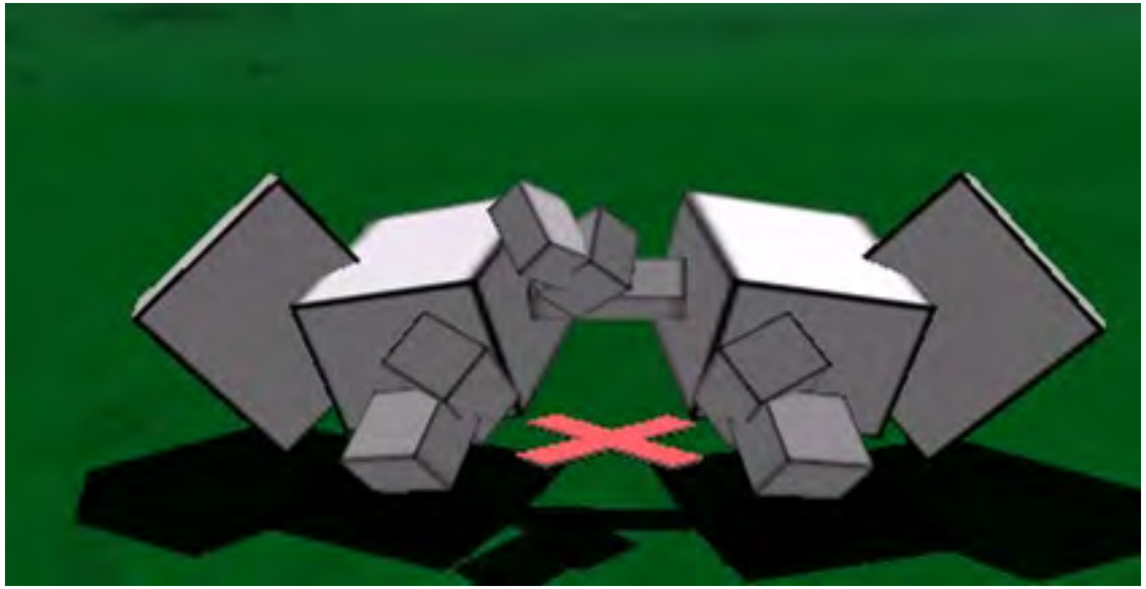


20 minutes to sexual maturity

3 years to sexual maturity

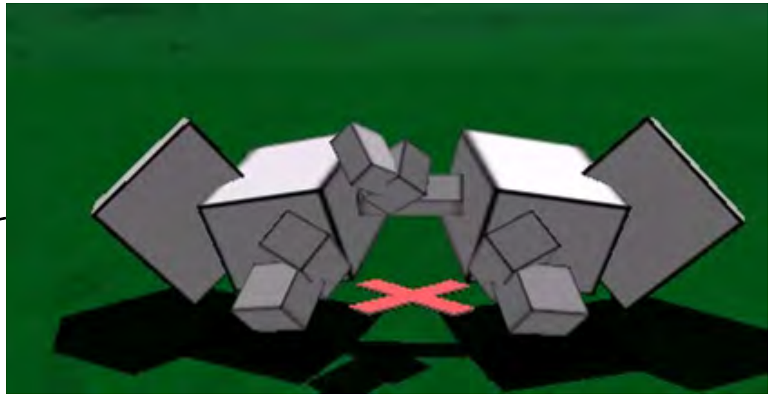
Illustrative Domain: Virtual Creatures

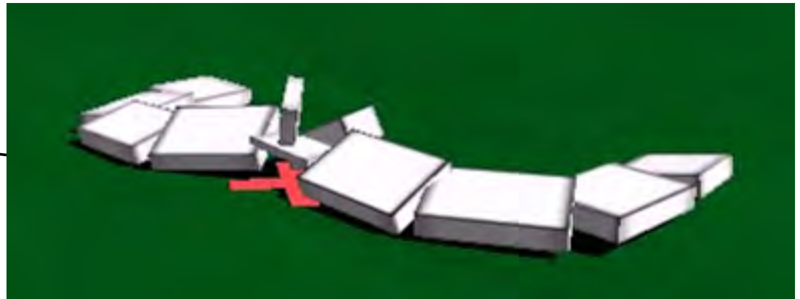
- Evolve both the morphology and controller of a virtual robot
- What if we want to see the best possible locomotion strategies for all areas of a morphology space?

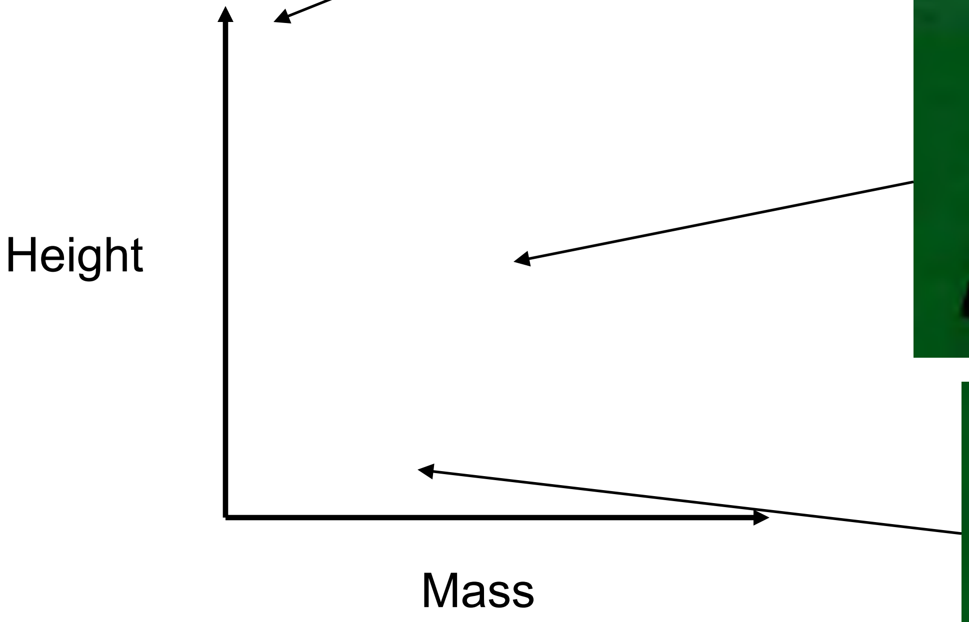


Morphology Space

- Height
- Mass
- Number of Active Joints



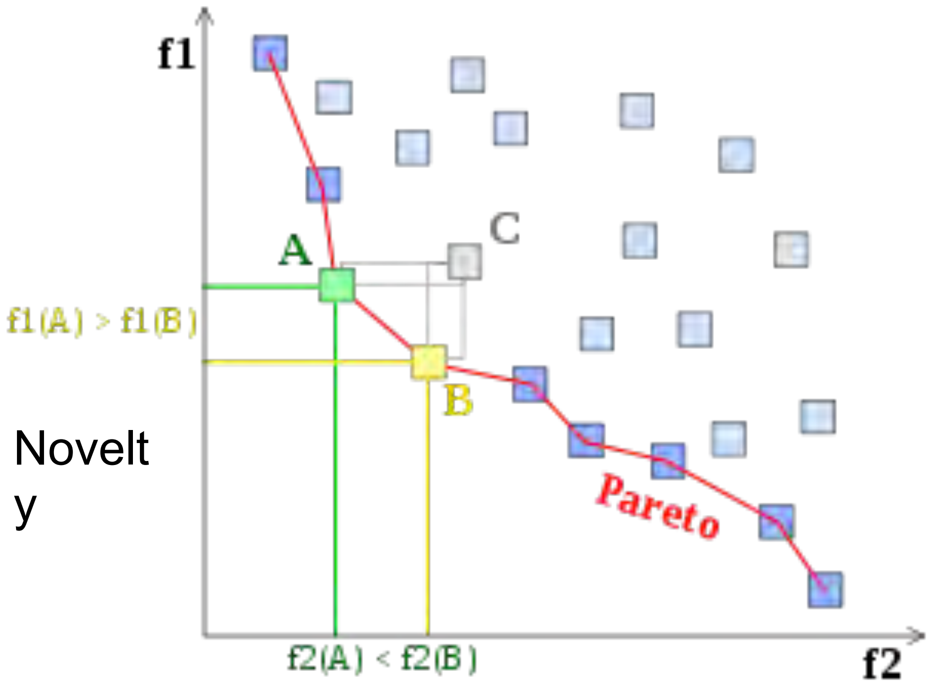




Novelty Search with Local Competition (Lehman and Stanley 2011)

- *Global* competition:
  Niches with higher capacity for objective performance favored
  - Compete globally on absolute performance score
- *Local competition*:
  Niches are explored relative to their local capacity for performance
  - Compete locally: how many of your morphological nearest-neighbors do you out-perform?

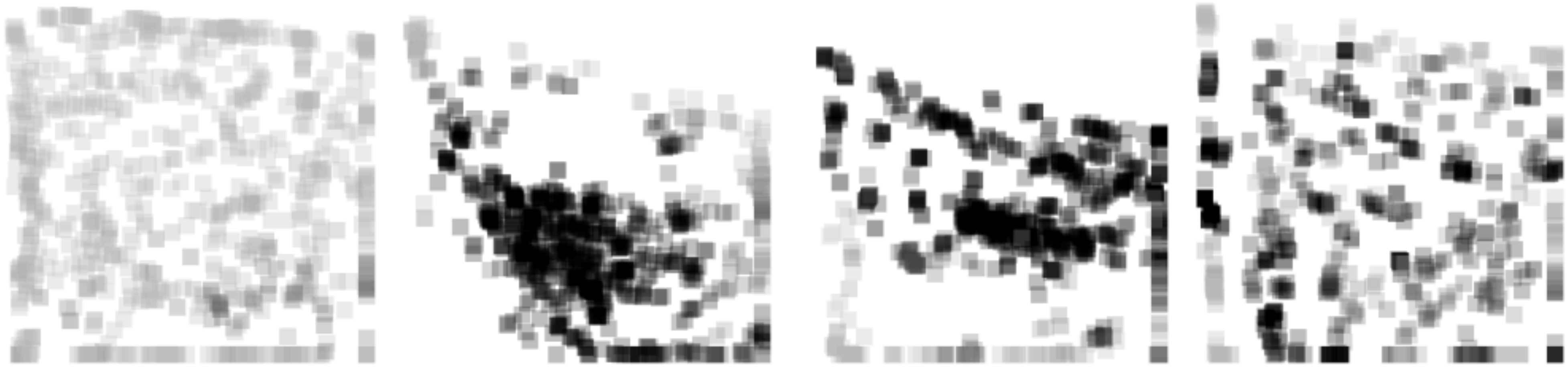
Novelty Search with Local Competition



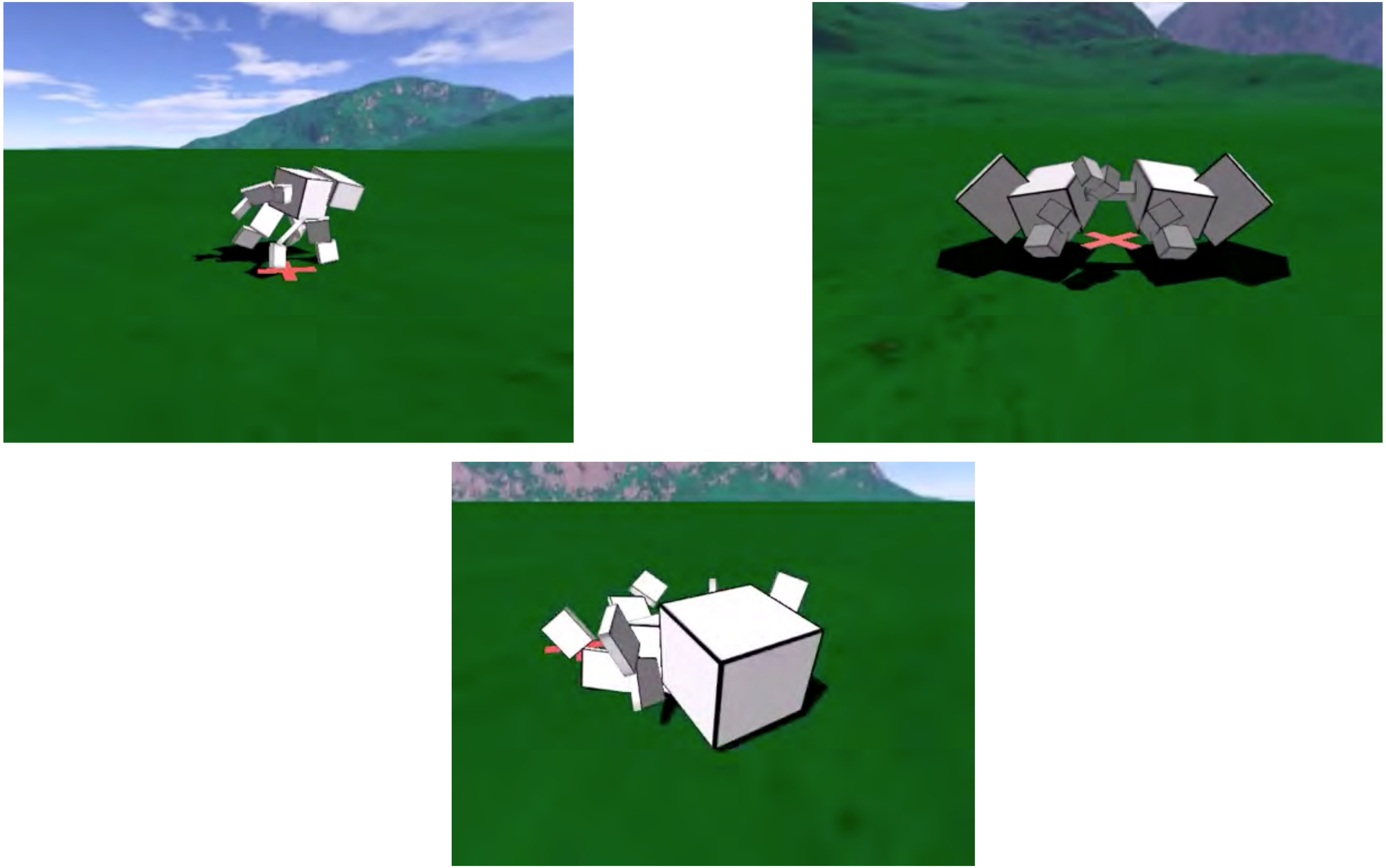
Novelt
y

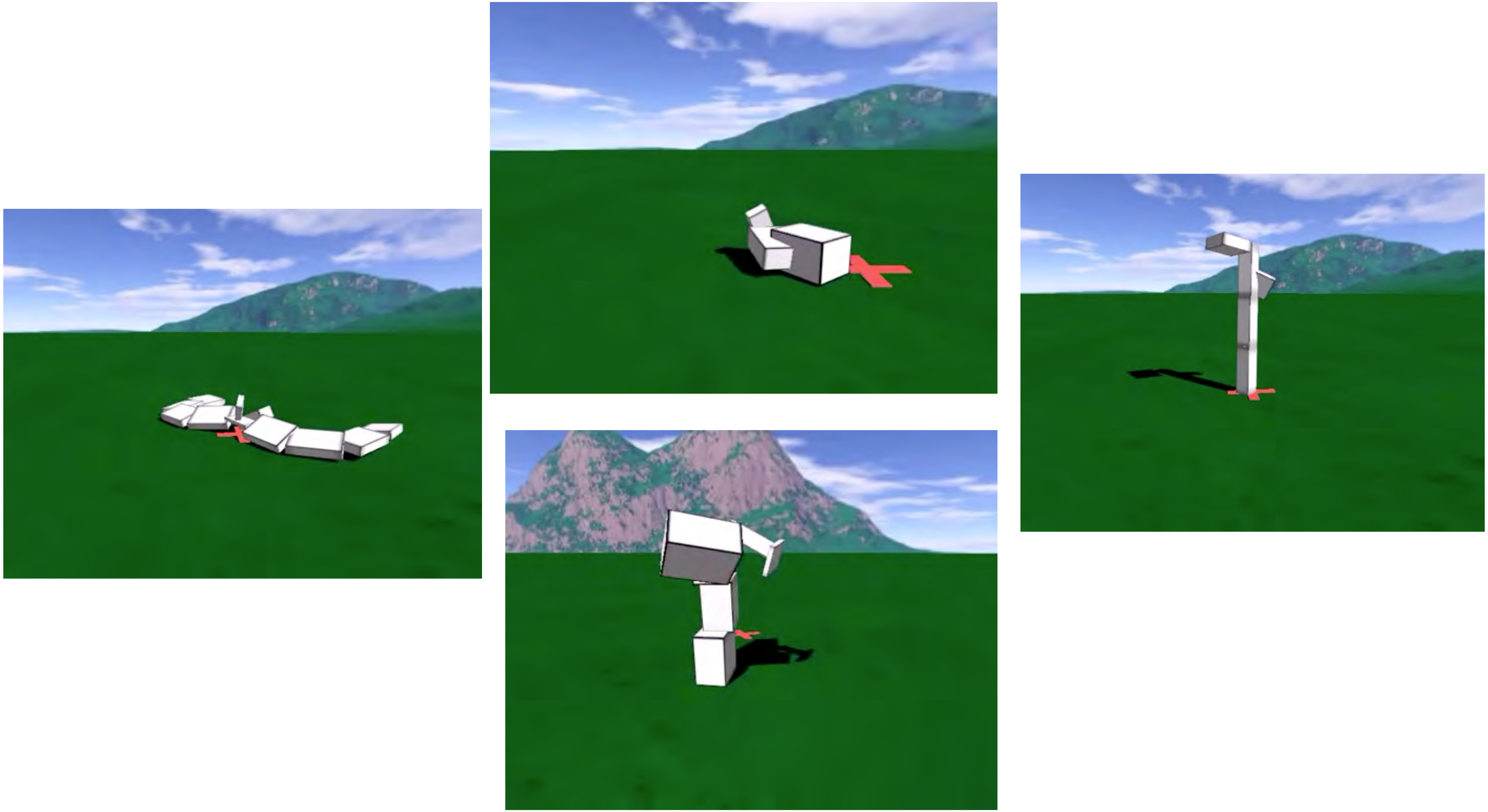
Local competition
score

Exploring the Morphology Space

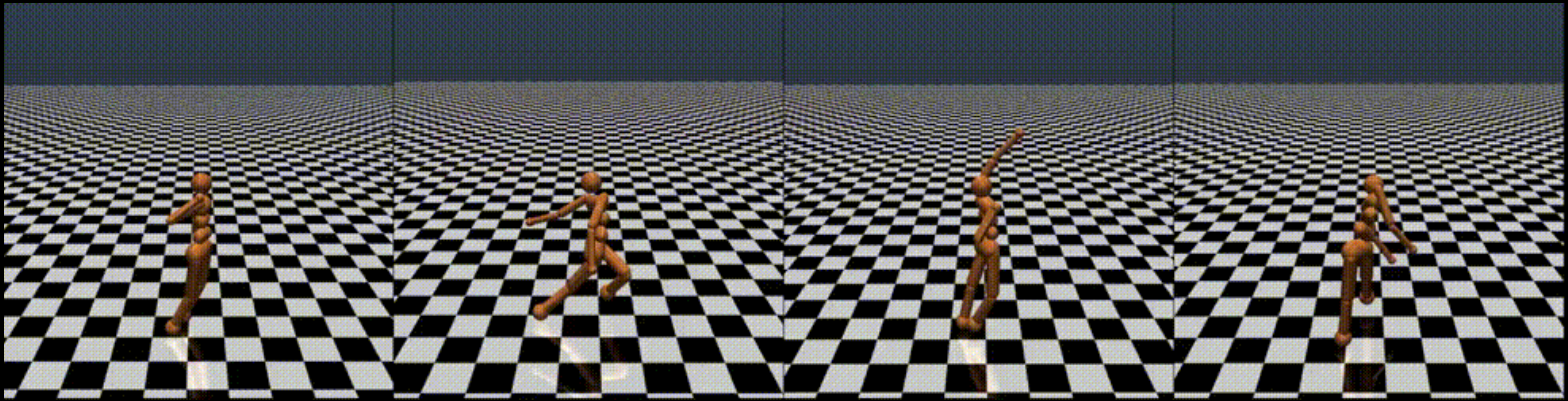


Novelty Objective Global Competition Local Competition



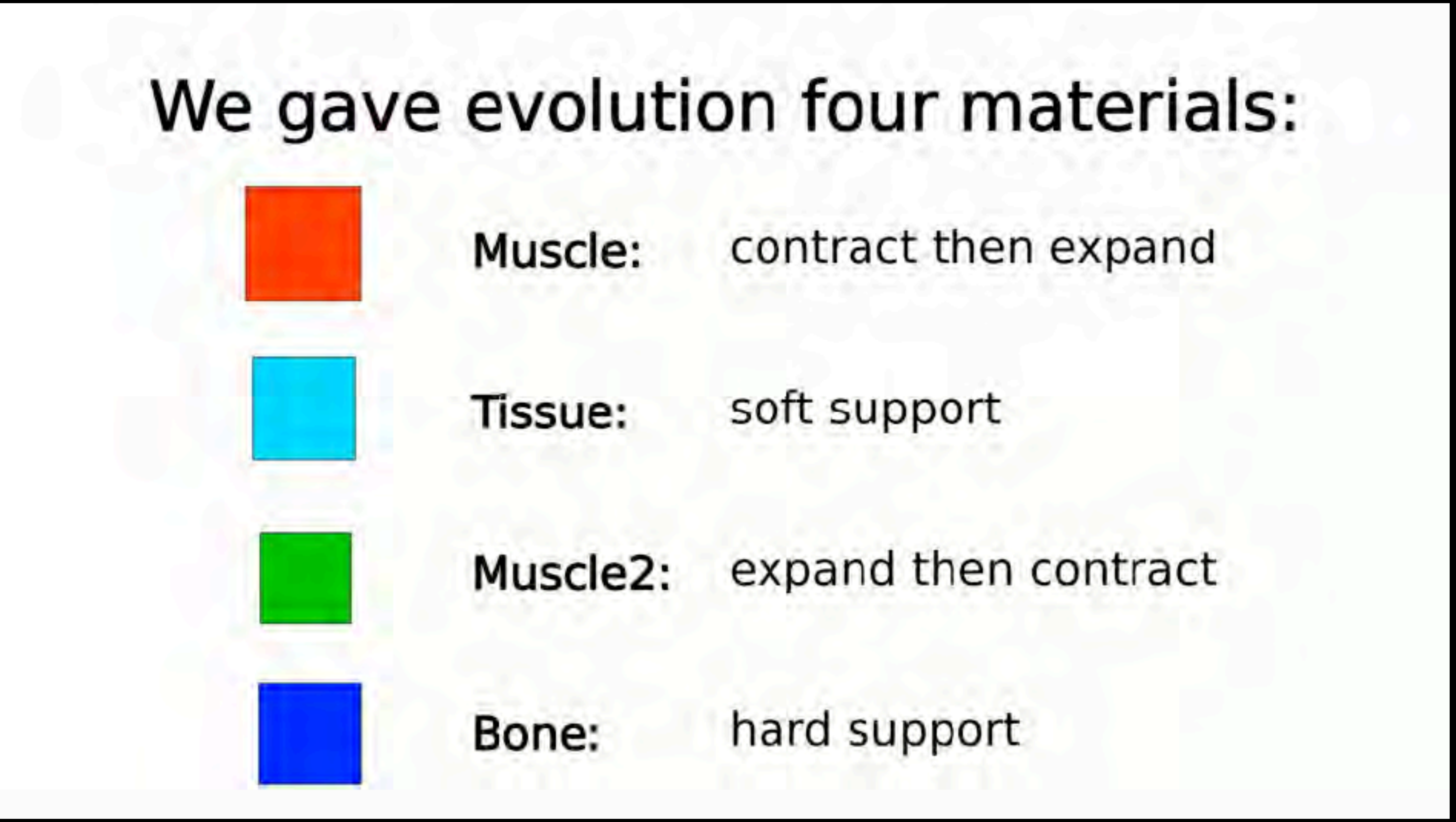


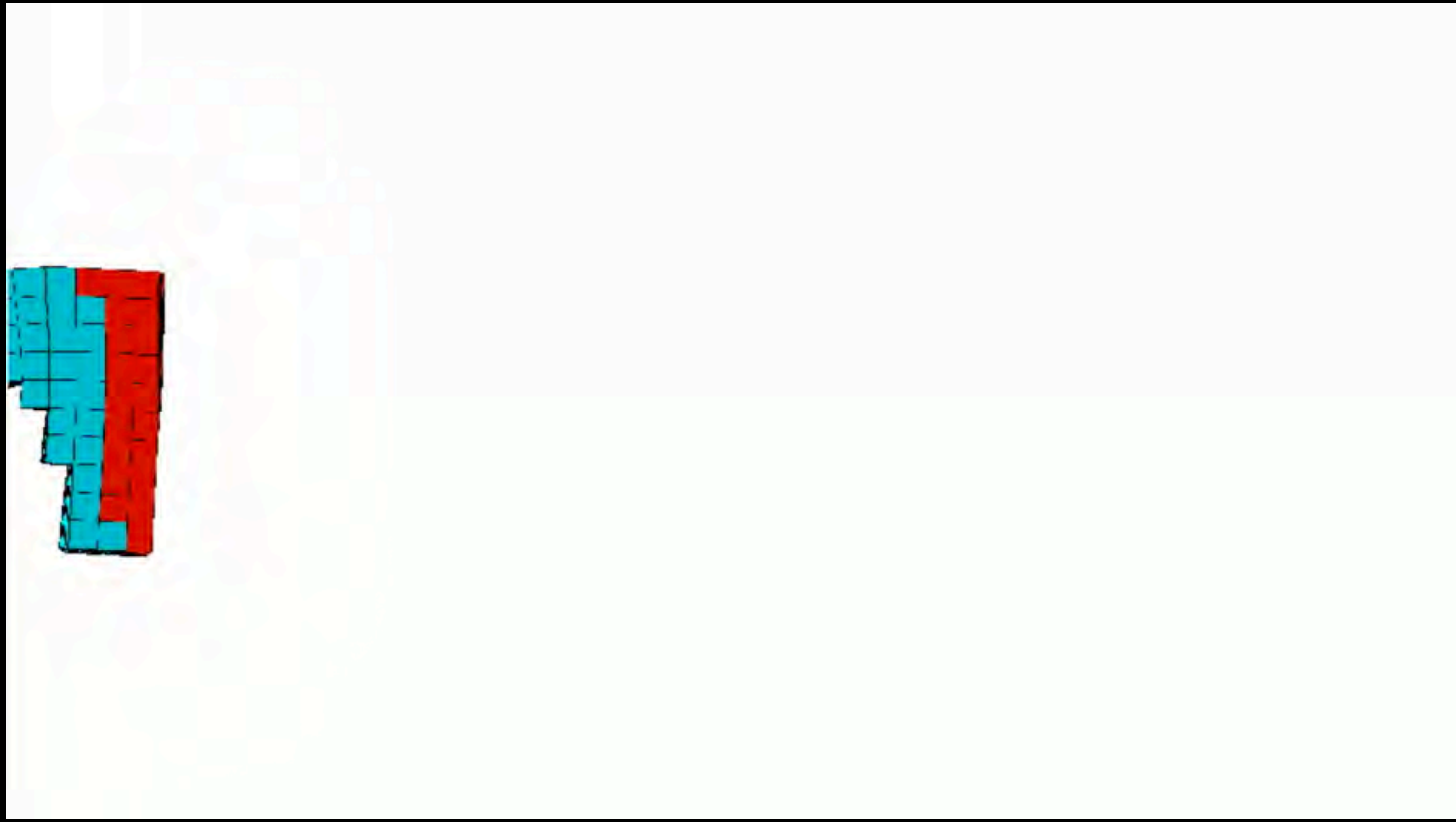
Traditional machine learning methods produce little diversity



Salimans, Ho, Chen, Sidor, Sutskever 2017

Population-based methods also produce little diversity





Cheney, MacCurdy, Clune, Lipson 2013

Quality Diversity Algorithms

- a diverse set of high-performing agents (policies)

Challenge: Diversity & Performance

- Quality diversity algorithms
  - Novelty Search + Local Competition (Lehman & Stanley)

Challenge: Diversity & Performance

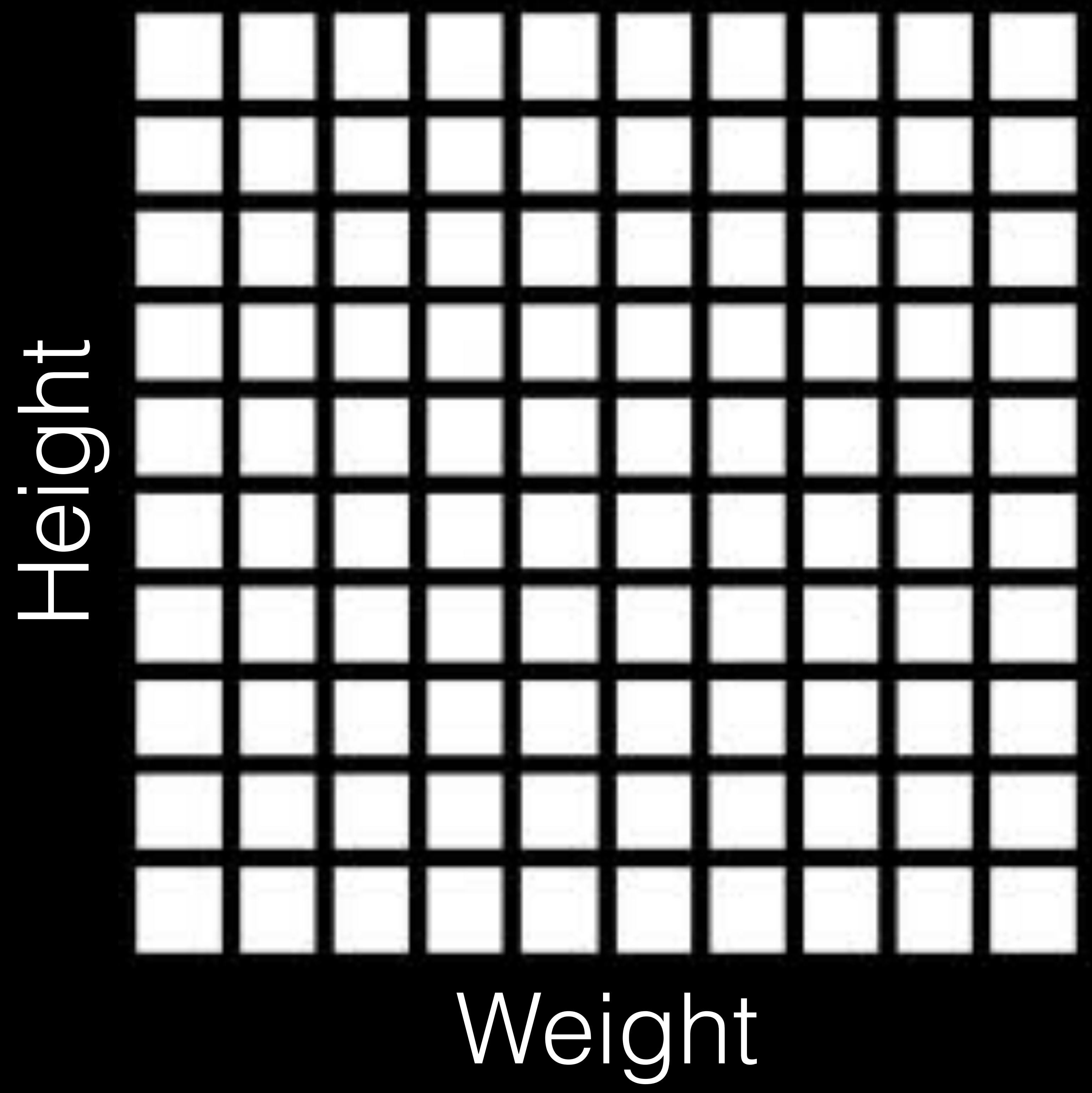
- Quality diversity algorithms
  - Novelty Search + Local Competition (Lehman & Stanley)
  - MAP-Elites (Mouret & Clune)

Jean-Baptiste Mouret

MAP-Elites

Mouret & Clune 2015

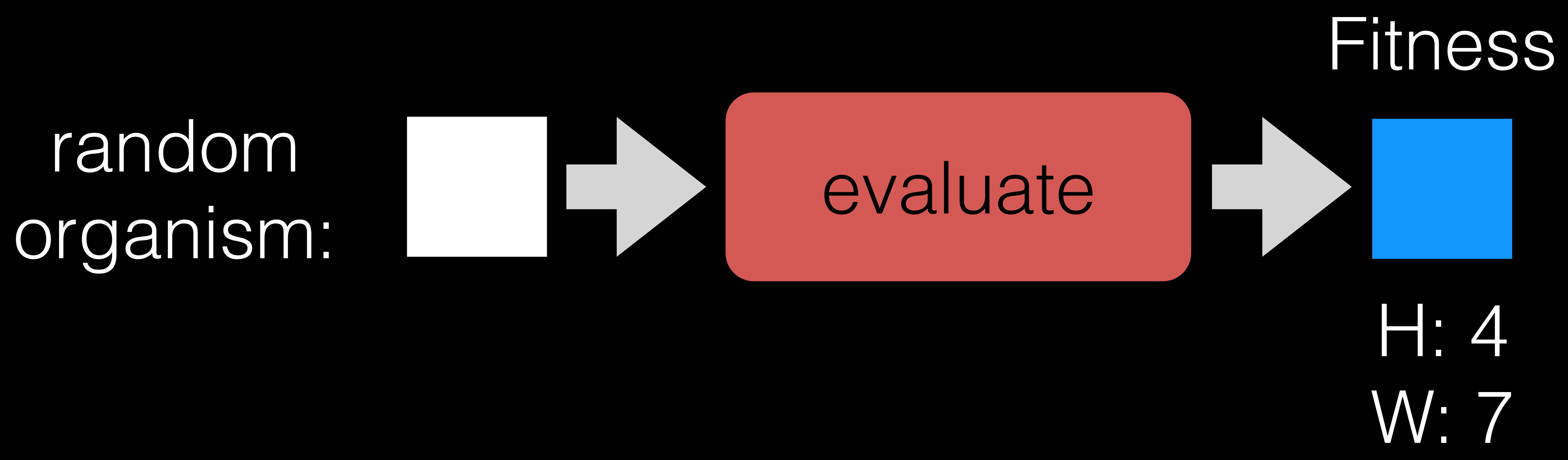
- Multi-dimensional Archive of Phenotypic Elites
  - Choose dimensions of interest in behavior space
  - Discretize
  - Mutate, locate, replace if better, repeat

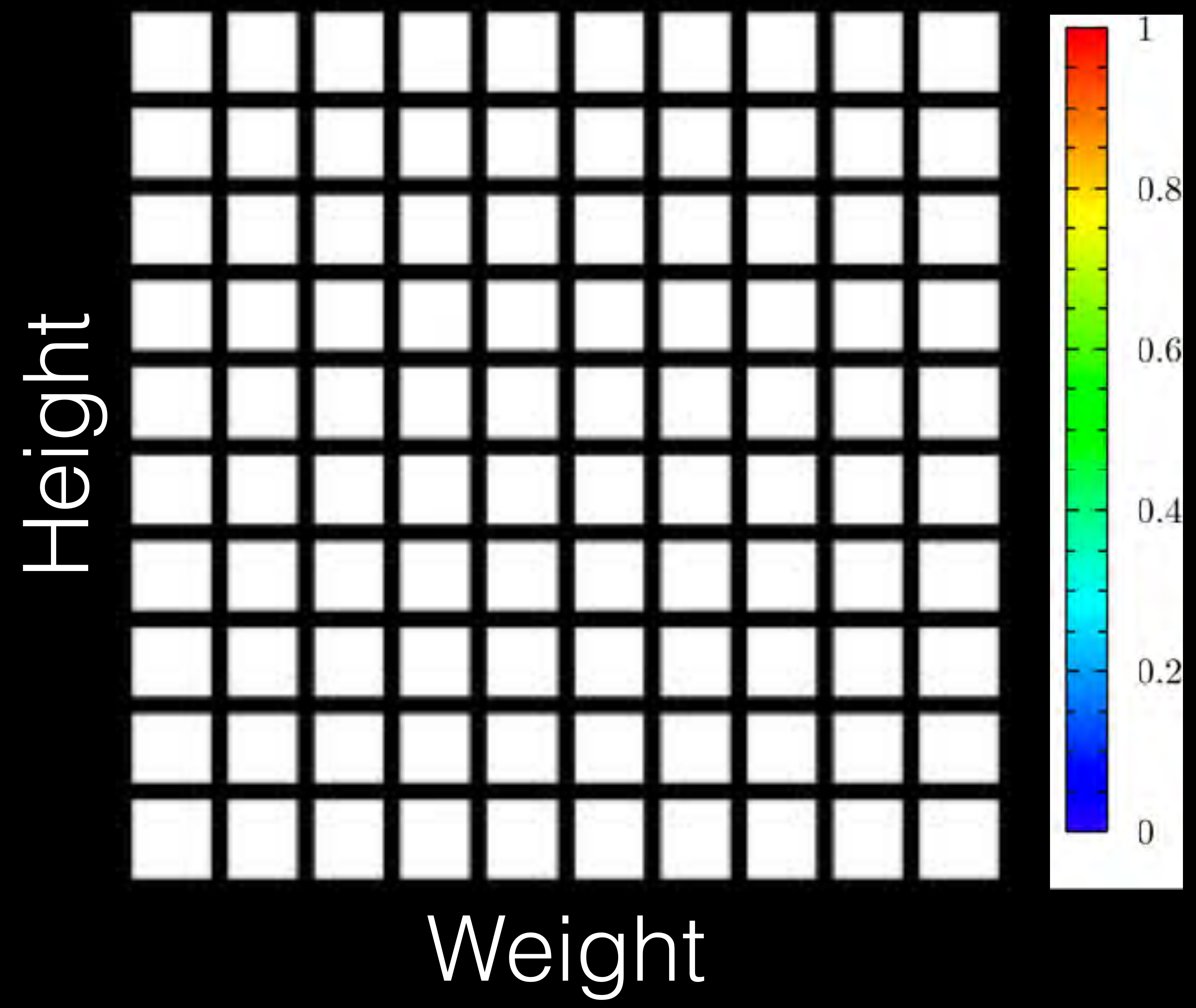


MAP-Elites

Mouret & Clune 2015

- Multi-dimensional Archive of Phenotypic Elites
  - Choose dimensions of interest in behavior space
  - Discretize
  - Mutate, locate, replace if better, repeat





MAP-Elites

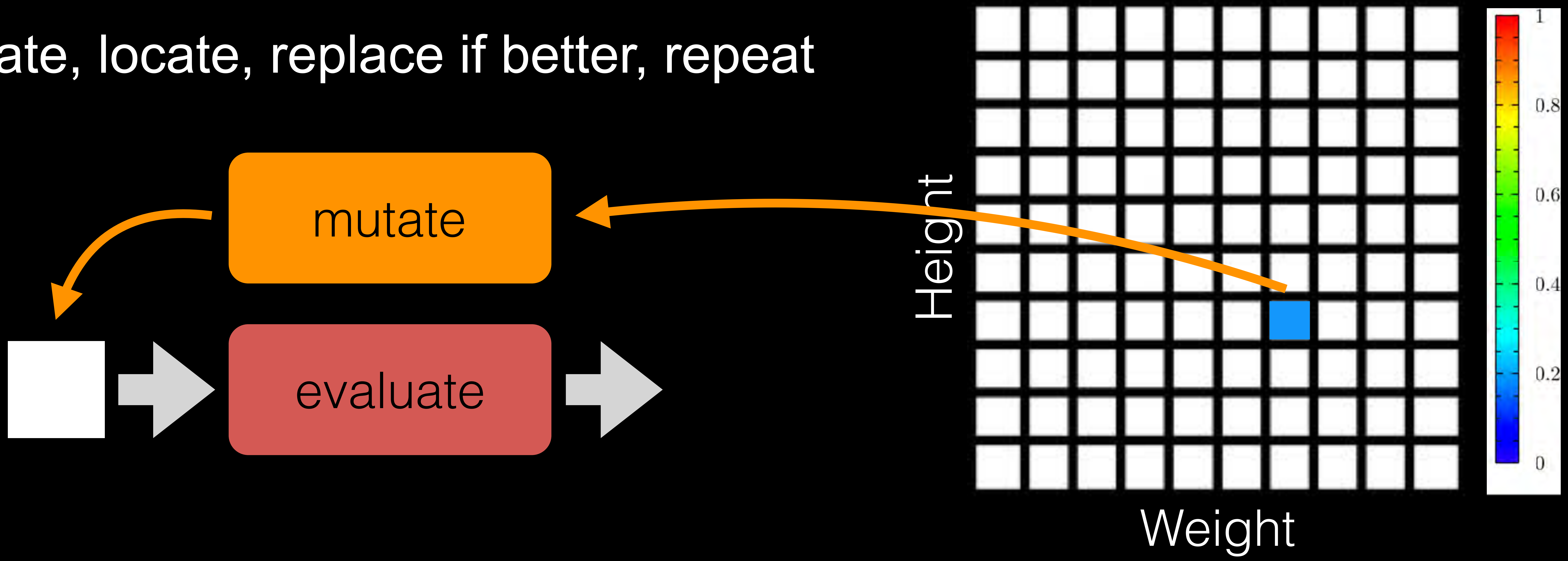
Mouret & Clune 2015

- Multi-dimensional Archive of Phenotypic Elites
  - Choose dimensions of interest in behavior space
  - Discretize
  - Mutate, locate, replace if better, repeat

MAP-Elites

Mouret & Clune 2015

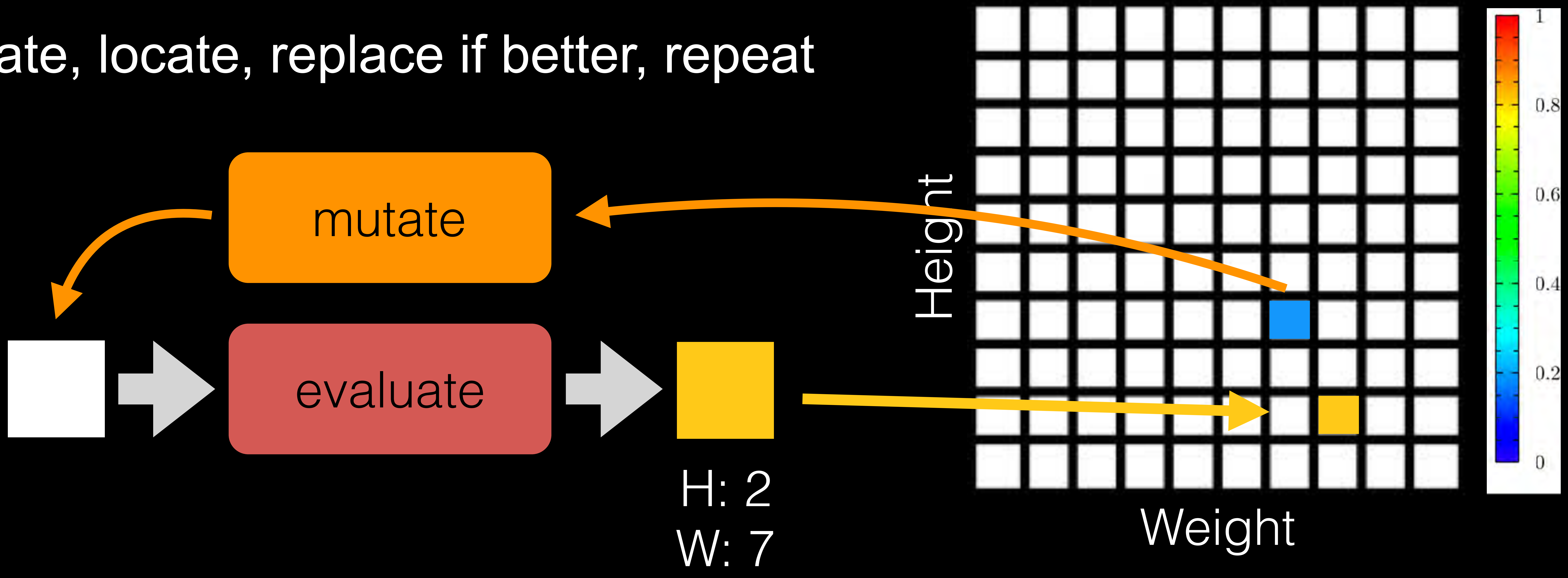
- Multi-dimensional Archive of Phenotypic Elites
  - Choose dimensions of interest in behavior space
  - Discretize
  - Mutate, locate, replace if better, repeat



MAP-Elites

Mouret & Clune 2015

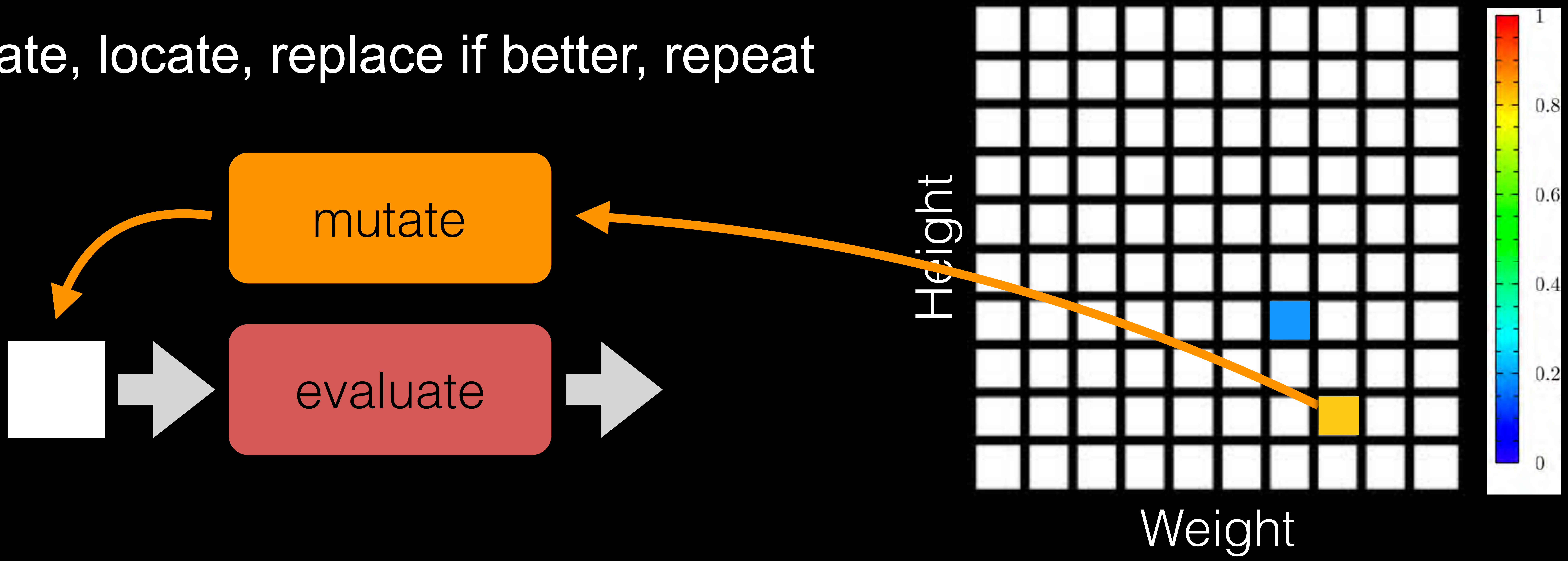
- Multi-dimensional Archive of Phenotypic Elites
  - Choose dimensions of interest in behavior space
  - Discretize
  - Mutate, locate, replace if better, repeat



MAP-Elites

Mouret & Clune 2015

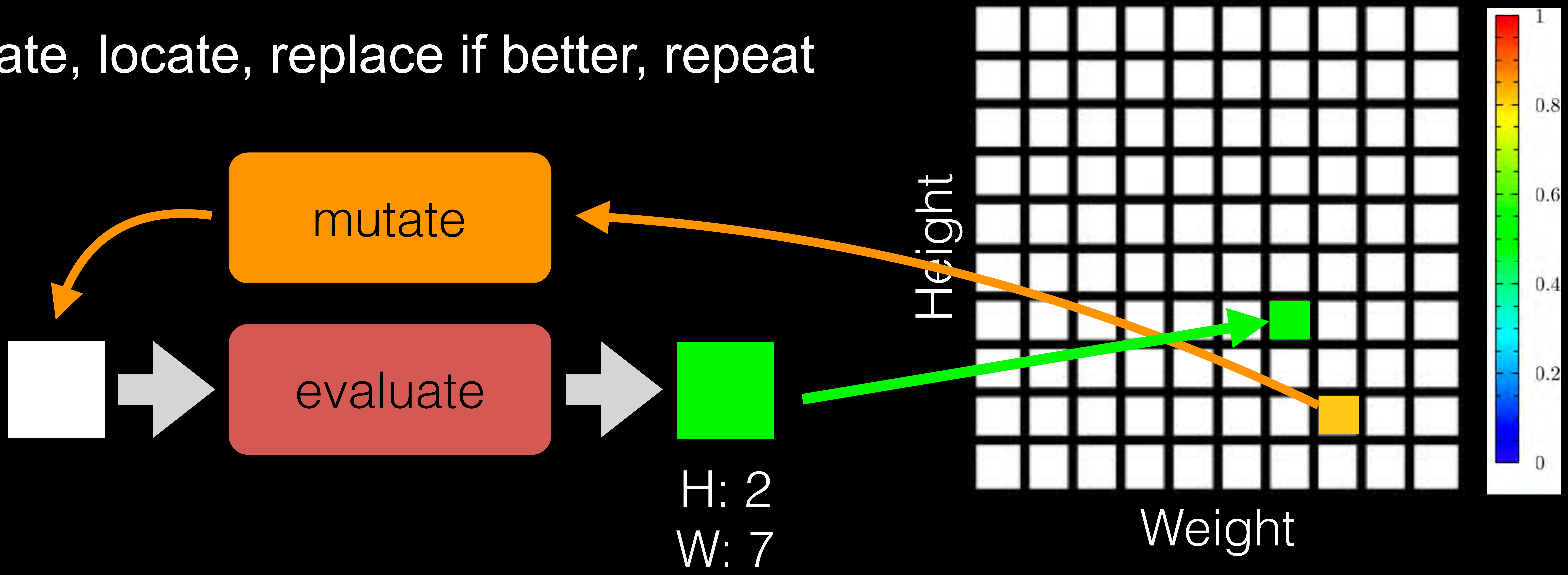
- Multi-dimensional Archive of Phenotypic Elites
  - Choose dimensions of interest in behavior space
  - Discretize
  - Mutate, locate, replace if better, repeat



MAP-Elites

Mouret & Clune 2015

- Multi-dimensional Archive of Phenotypic Elites
  - Choose dimensions of interest in behavior space
  - Discretize
  - Mutate, locate, replace if better, repeat

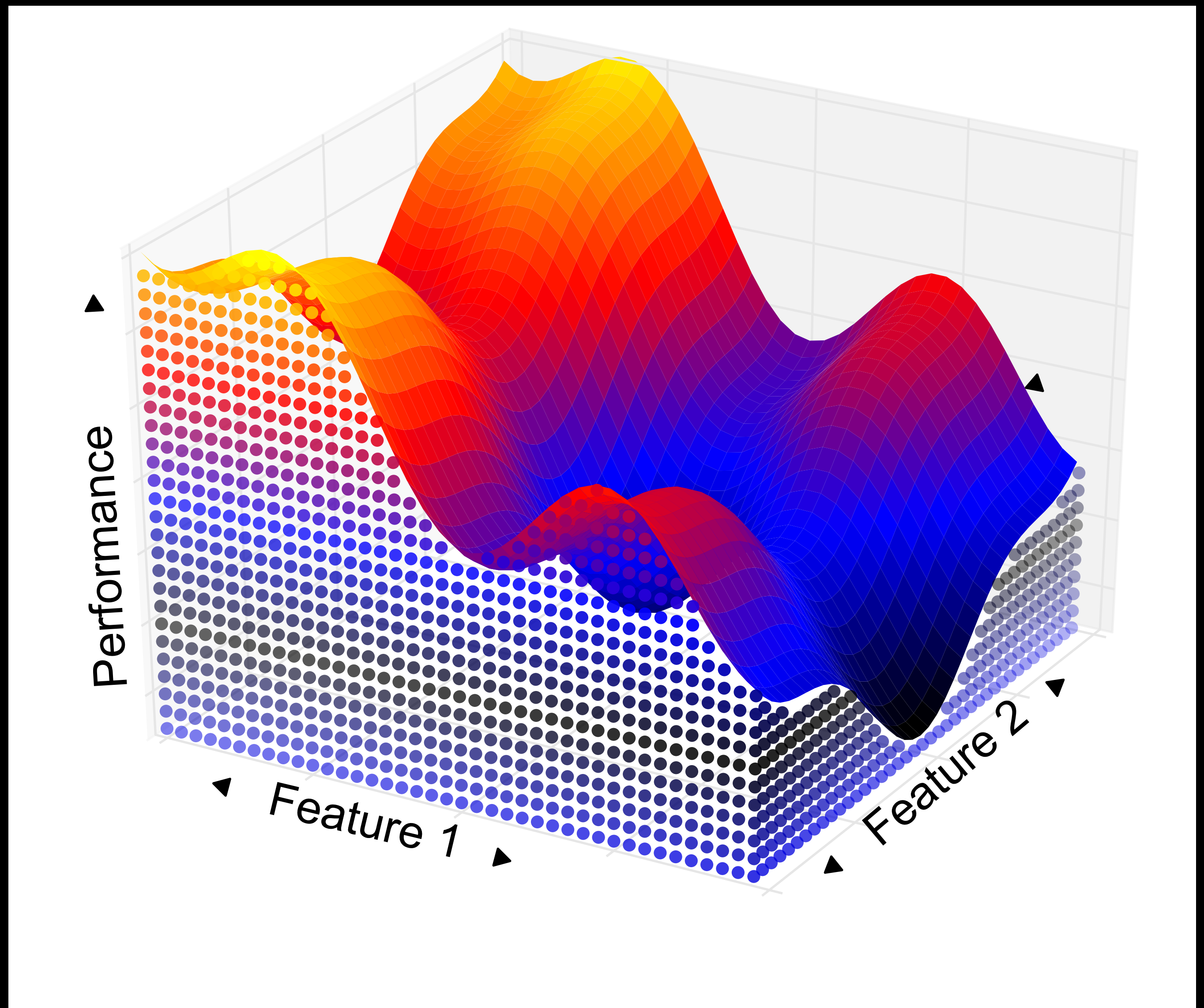


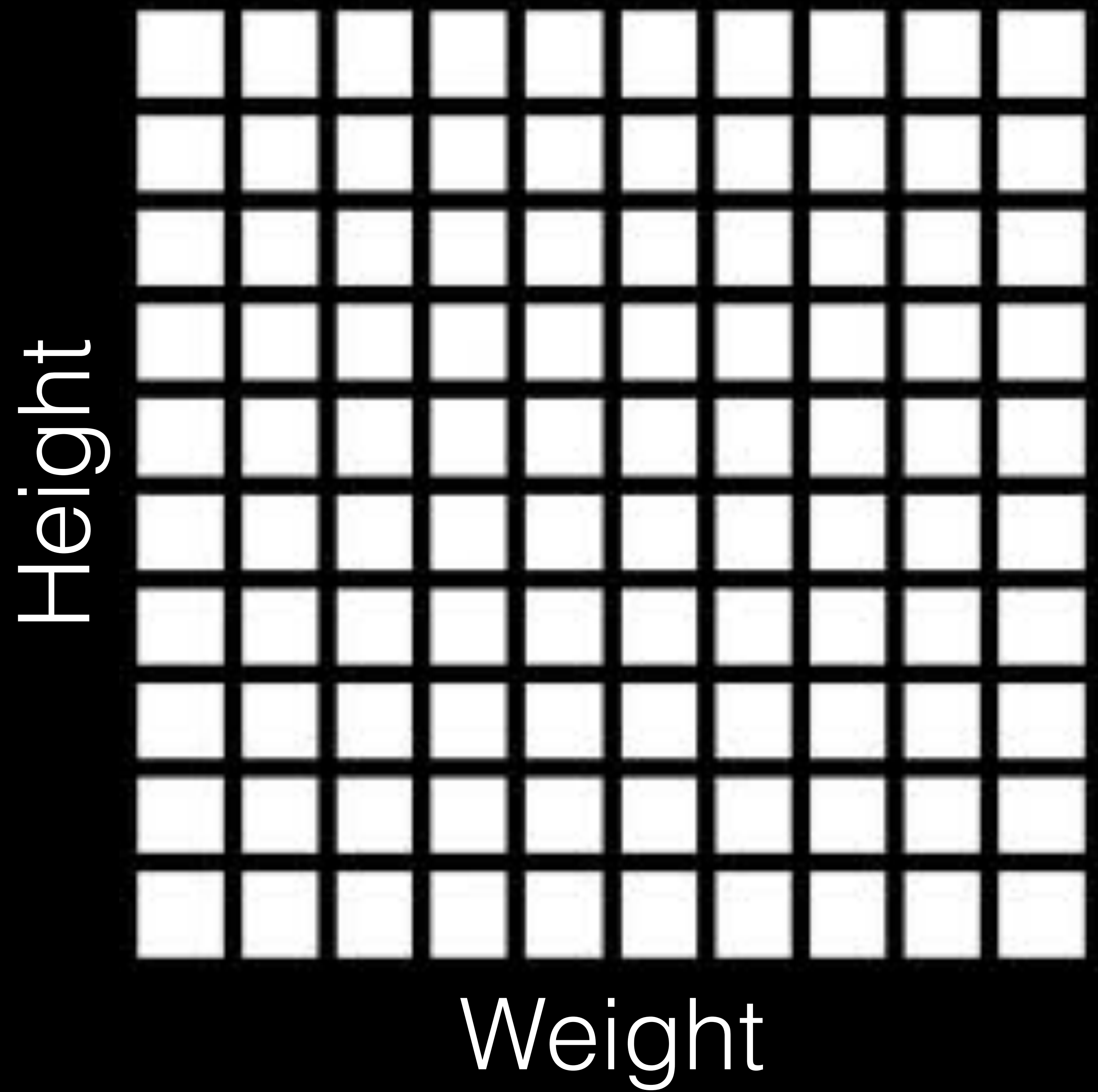
MAP-Elites

Mouret & Clune 2015

- Multi-dimensional Archive of Phenotypic Elites
  - Choose dimensions of interest in behavior space
  - Discretize
  - Mutate, locate, replace if better, repeat

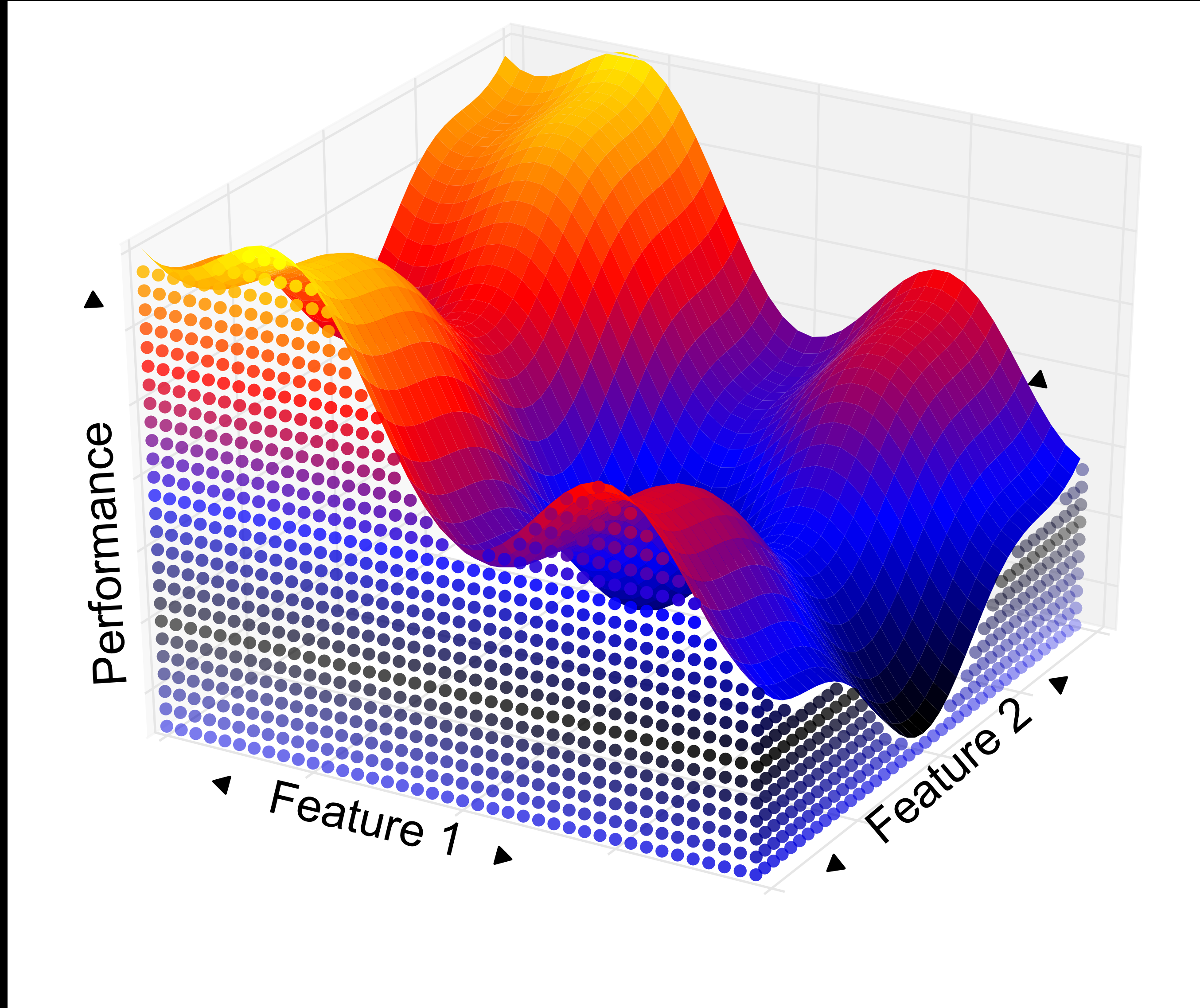
Set of diverse, high-quality solutions



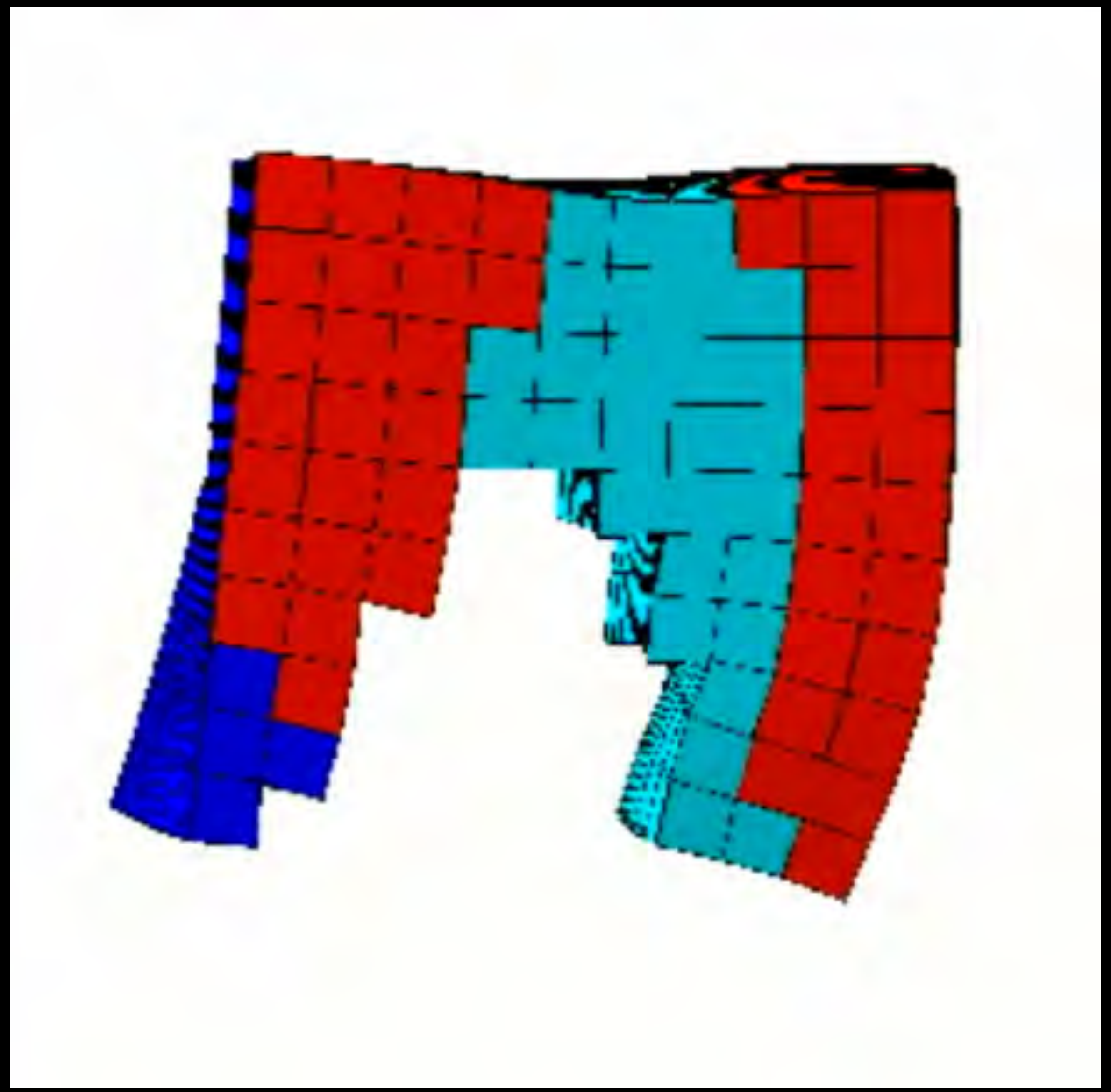


Soft Robots Problem

Mouret & Clune 2015



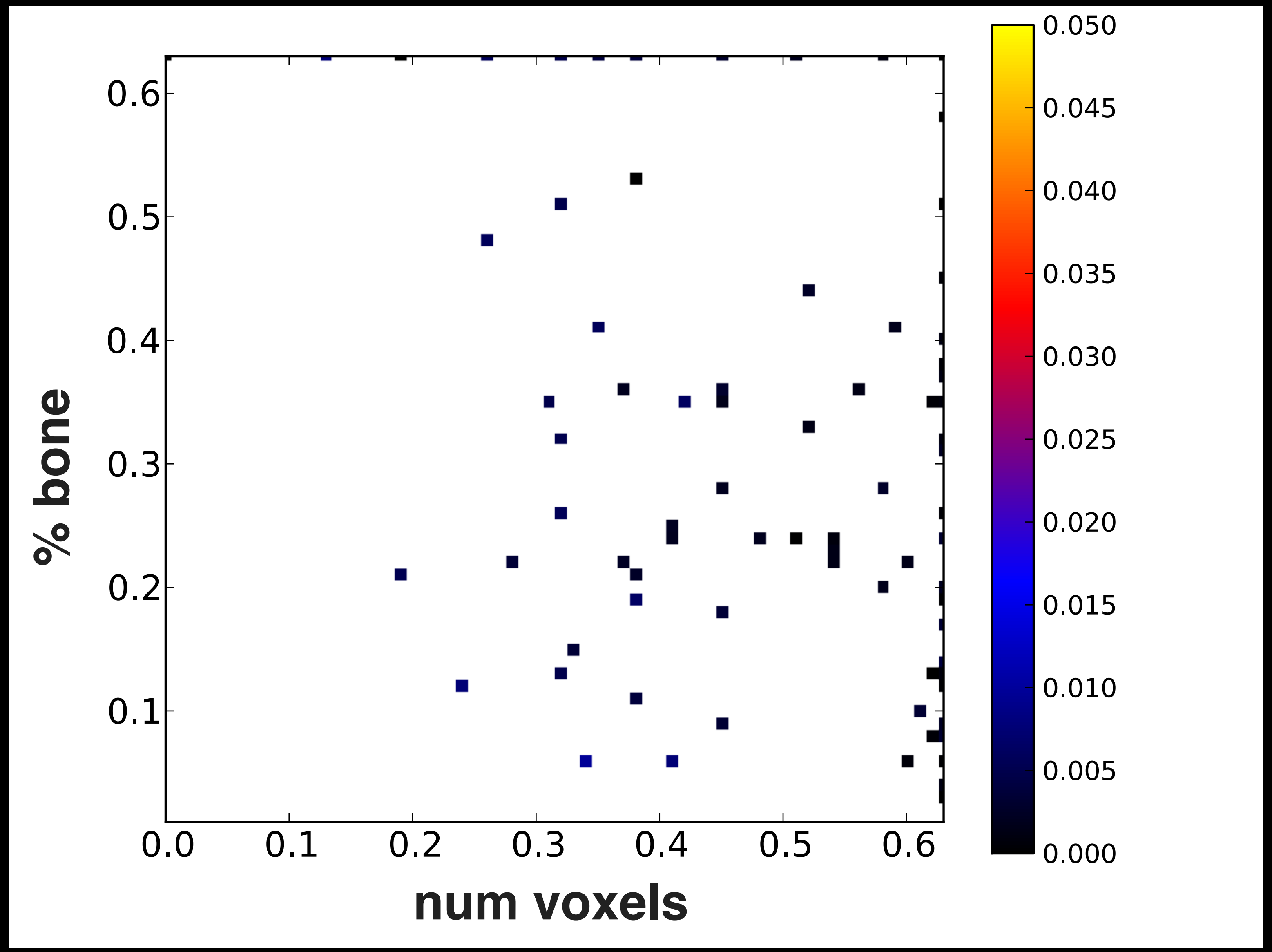
- Dimensions
  - number of voxels
  - % bone (dark blue)



Soft Robots Problem

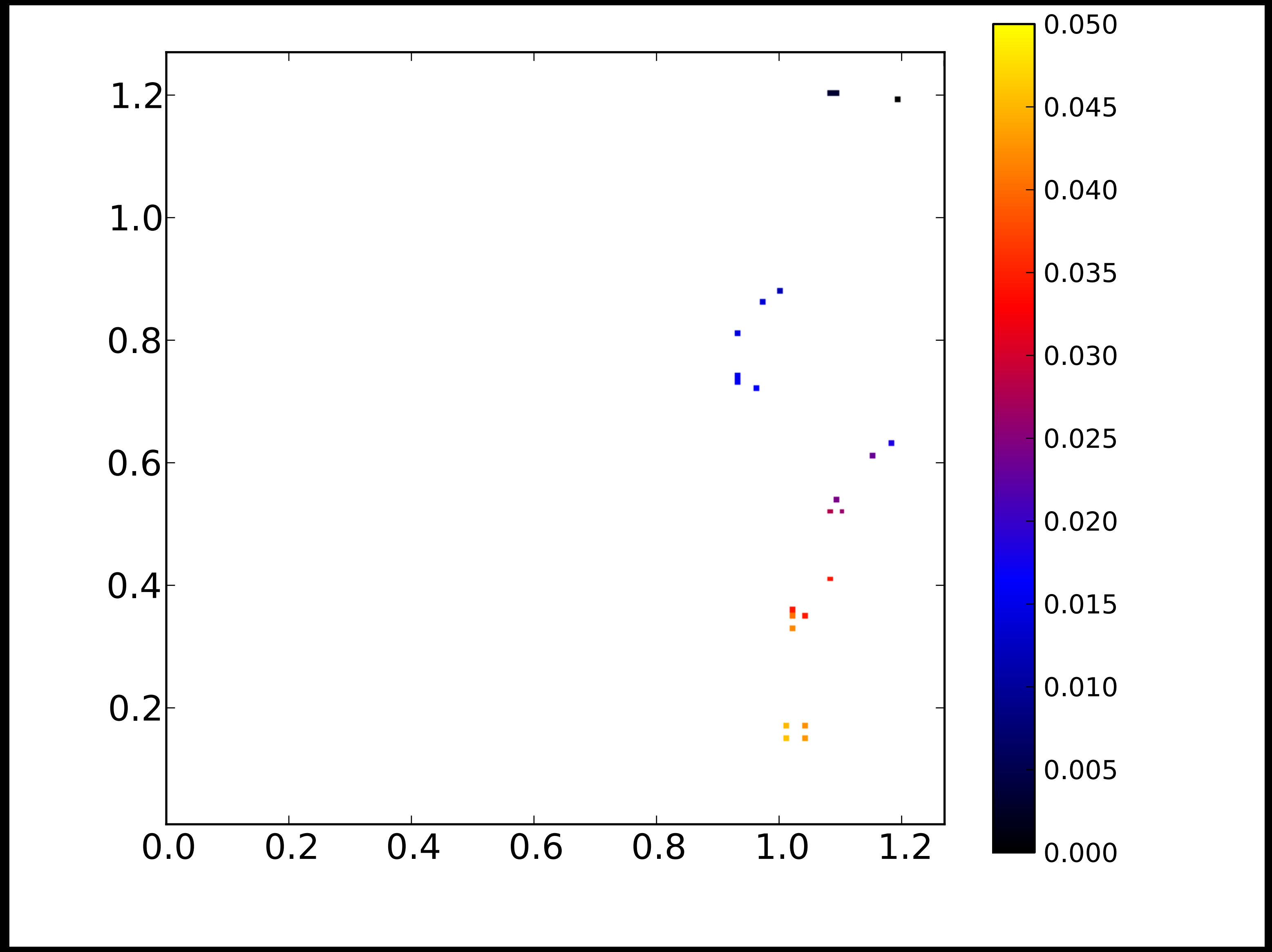
Mouret & Clune 2015

Classic Optimization



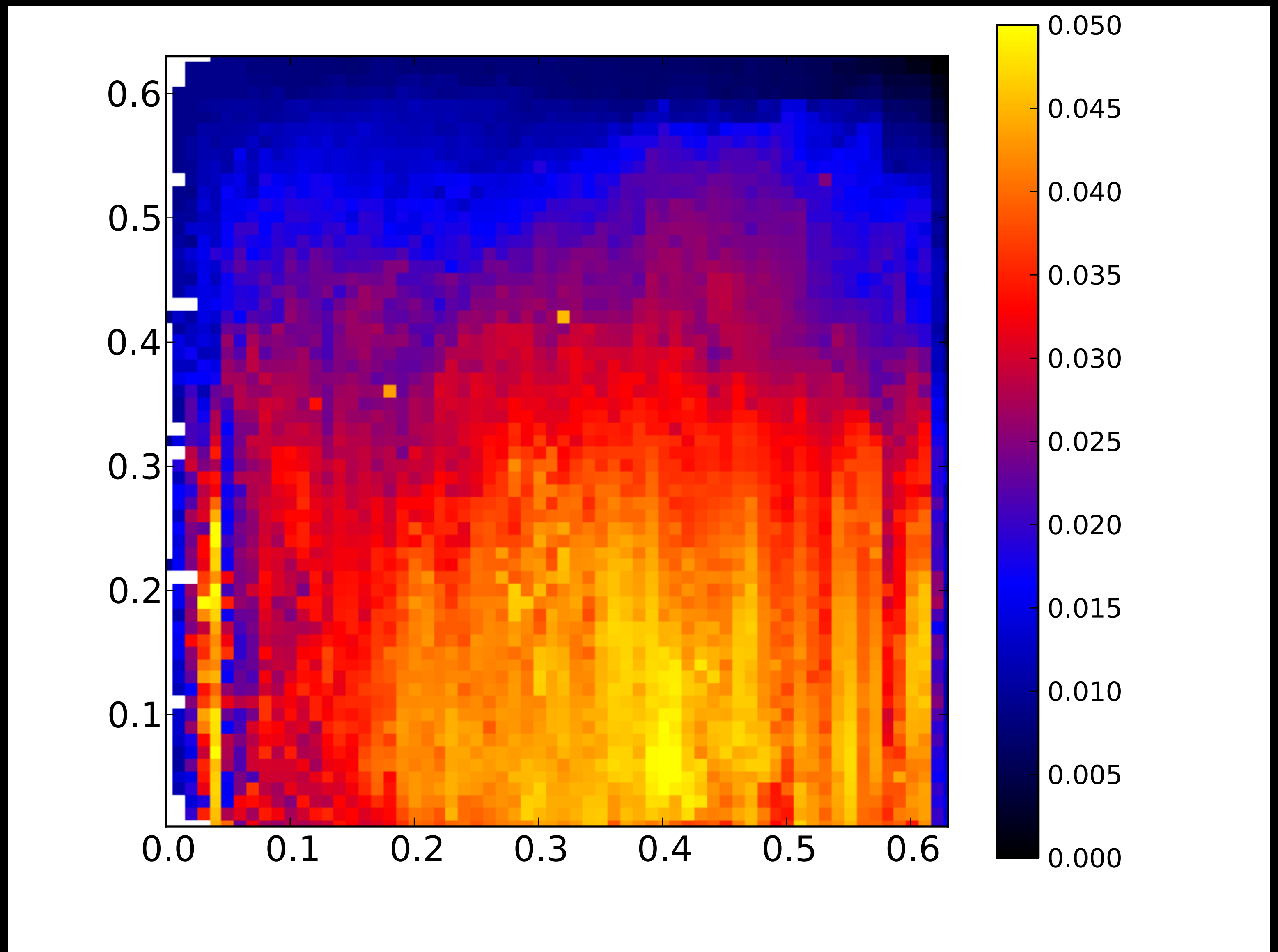
EA

Classic + Diversity

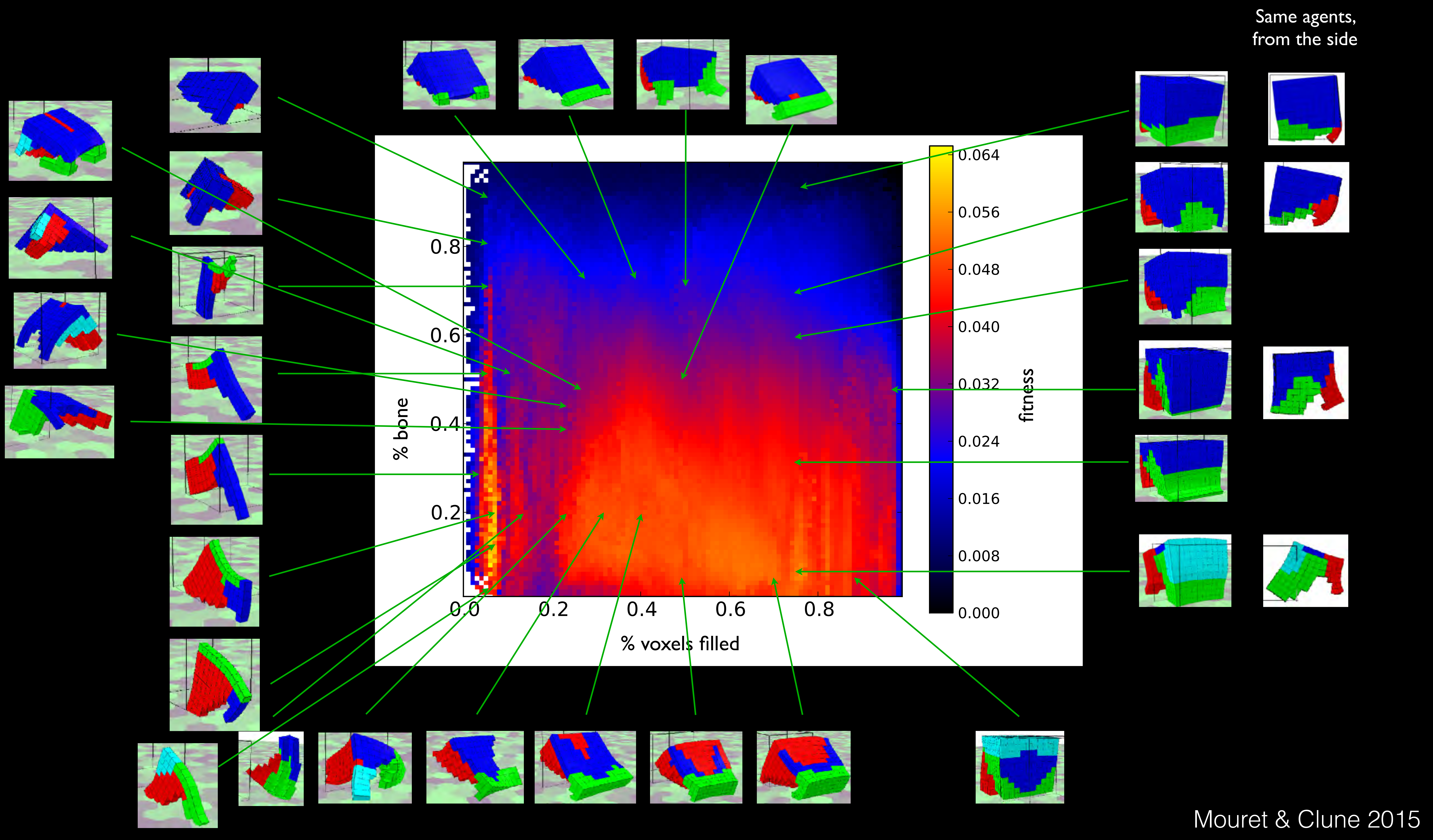


multi-objective EA

MAP-Elites



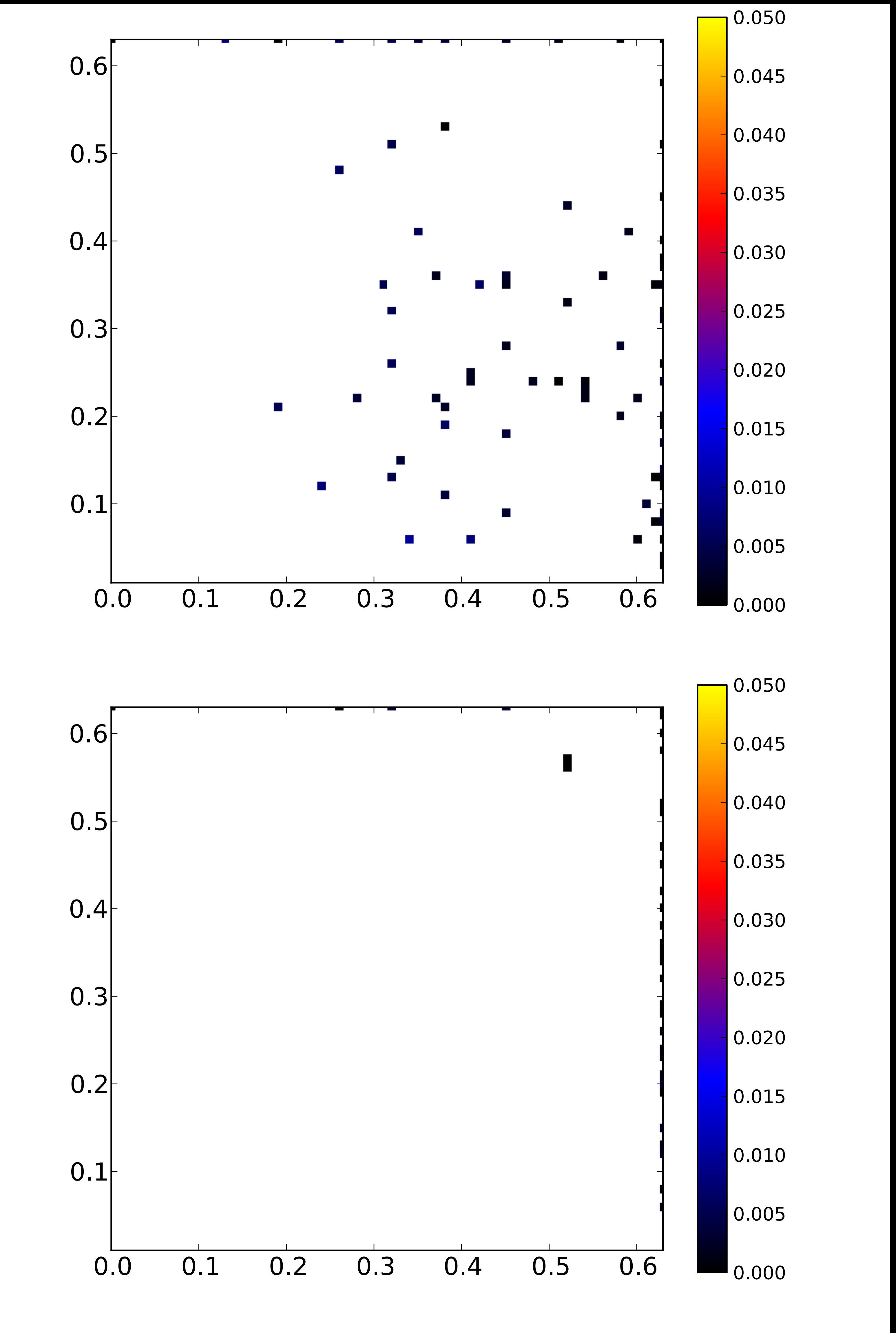
same # evals!



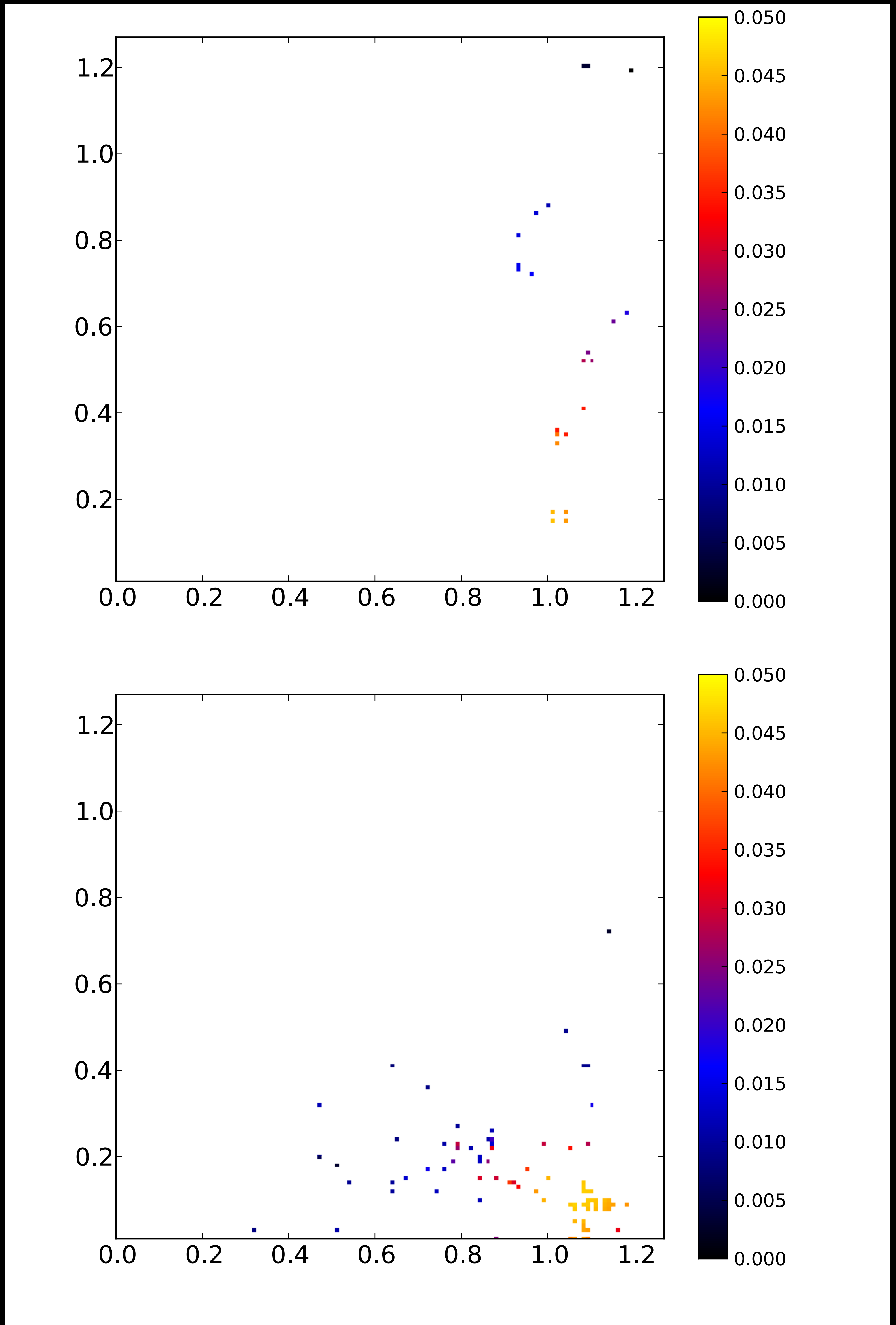
Same agents, from the side
% bone
% voxels filled
fitness
0.0
0.2
0.4
0.6
0.8
0.000
0.008
0.016
0.024
0.032
0.040
0.048
0.056
0.064
Mouret & Clune 2015

Different Runs: Soft Robot Problem

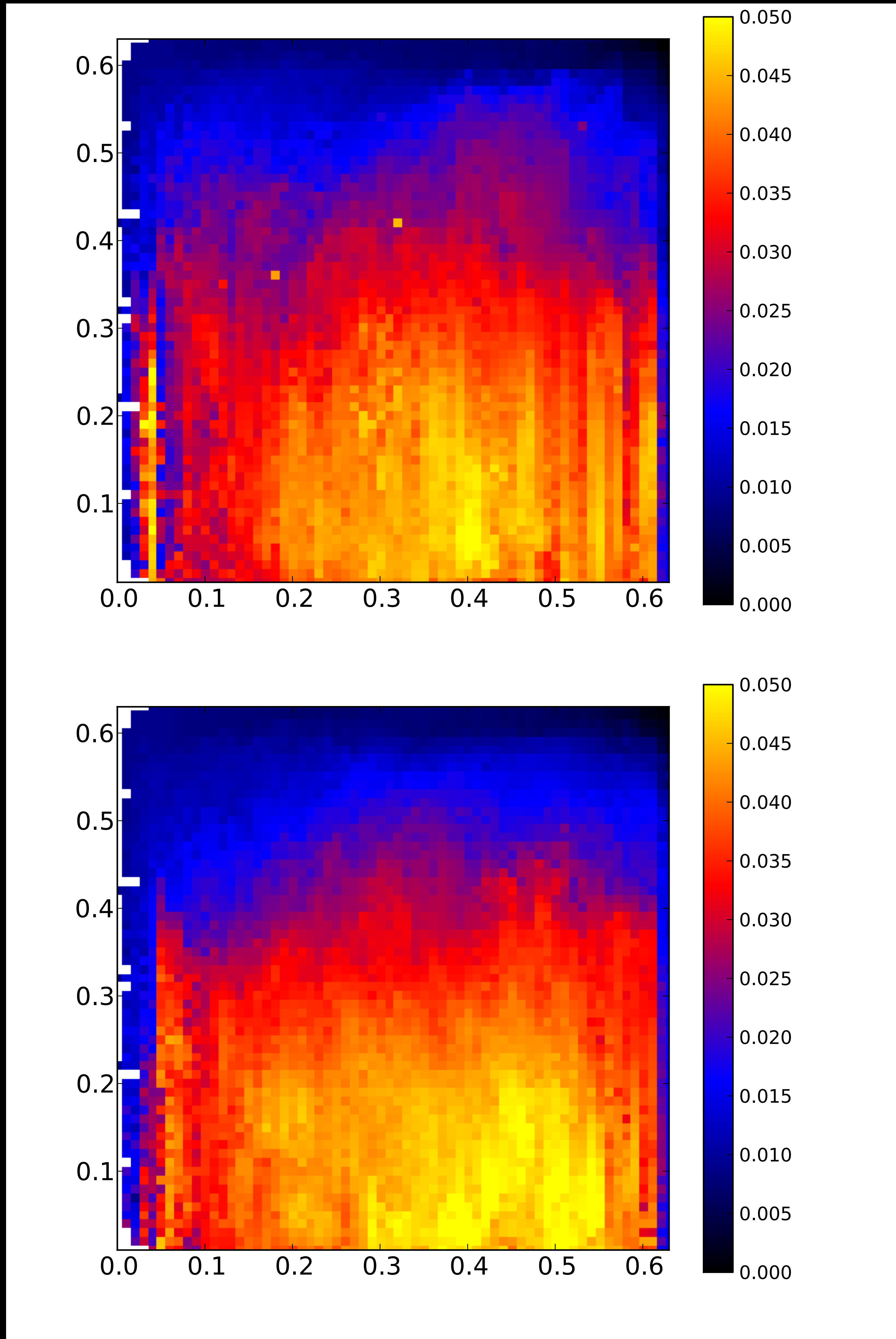
Classic Optimization

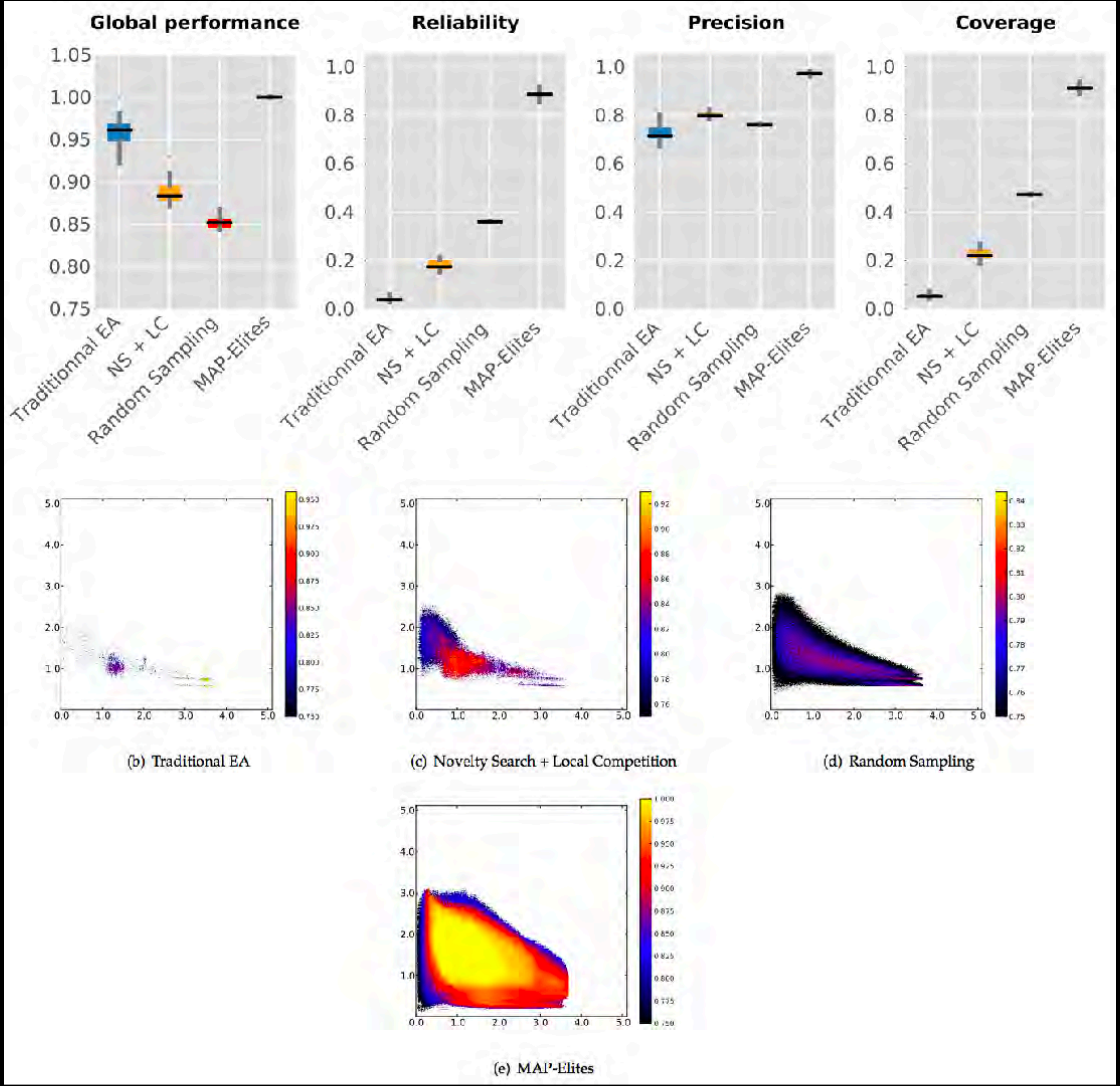


Classic + Diversity



MAP-Elites





(b) Traditional EA
(c) Novelty Search + Local Competition
(d) Random Sampling
(e) MAP-Elites

Retina Problem

Mouret & Clune 2015

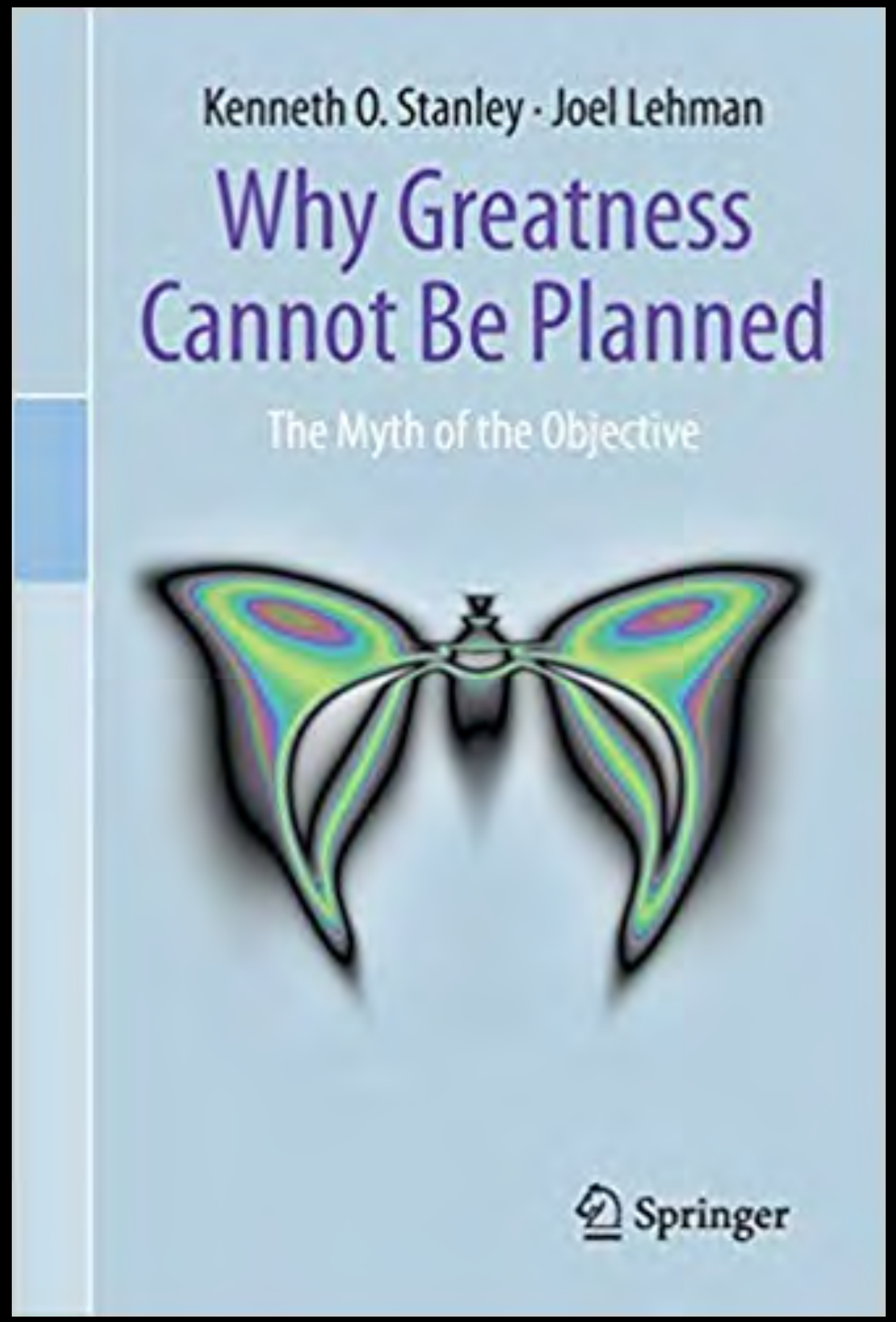
“Goal Switching”

Nguyen, Yosinski & Clune 2016

- When trying to solve task A, if you make progress on task B
  - keep the innovation and let it keep working on B

Goal Switching:
Key for Science & Technological Innovation

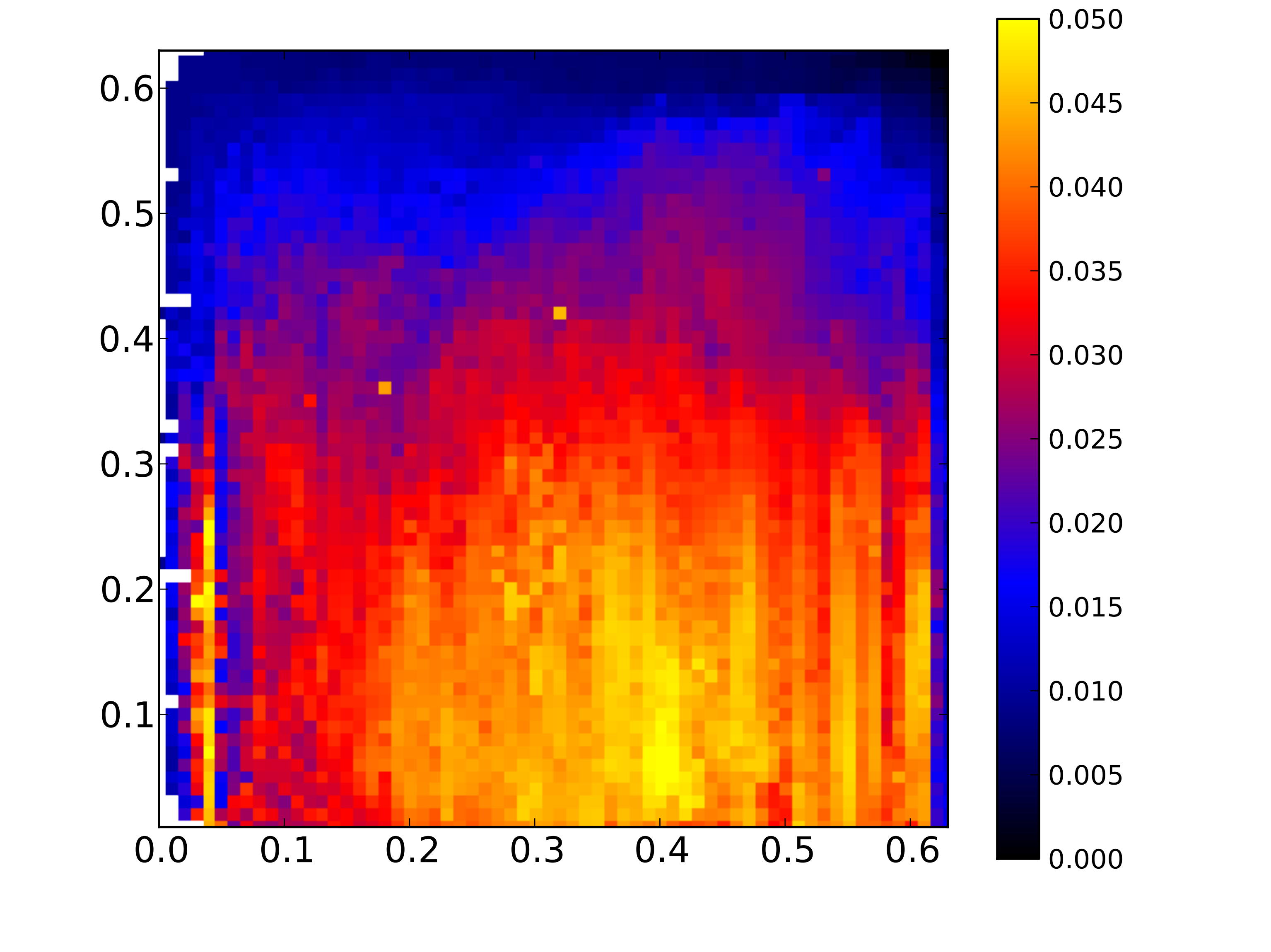
- Radar → microwaves
- Vacuum tubes → computers
- basic physics → clean energy (nuclear)
- etc.



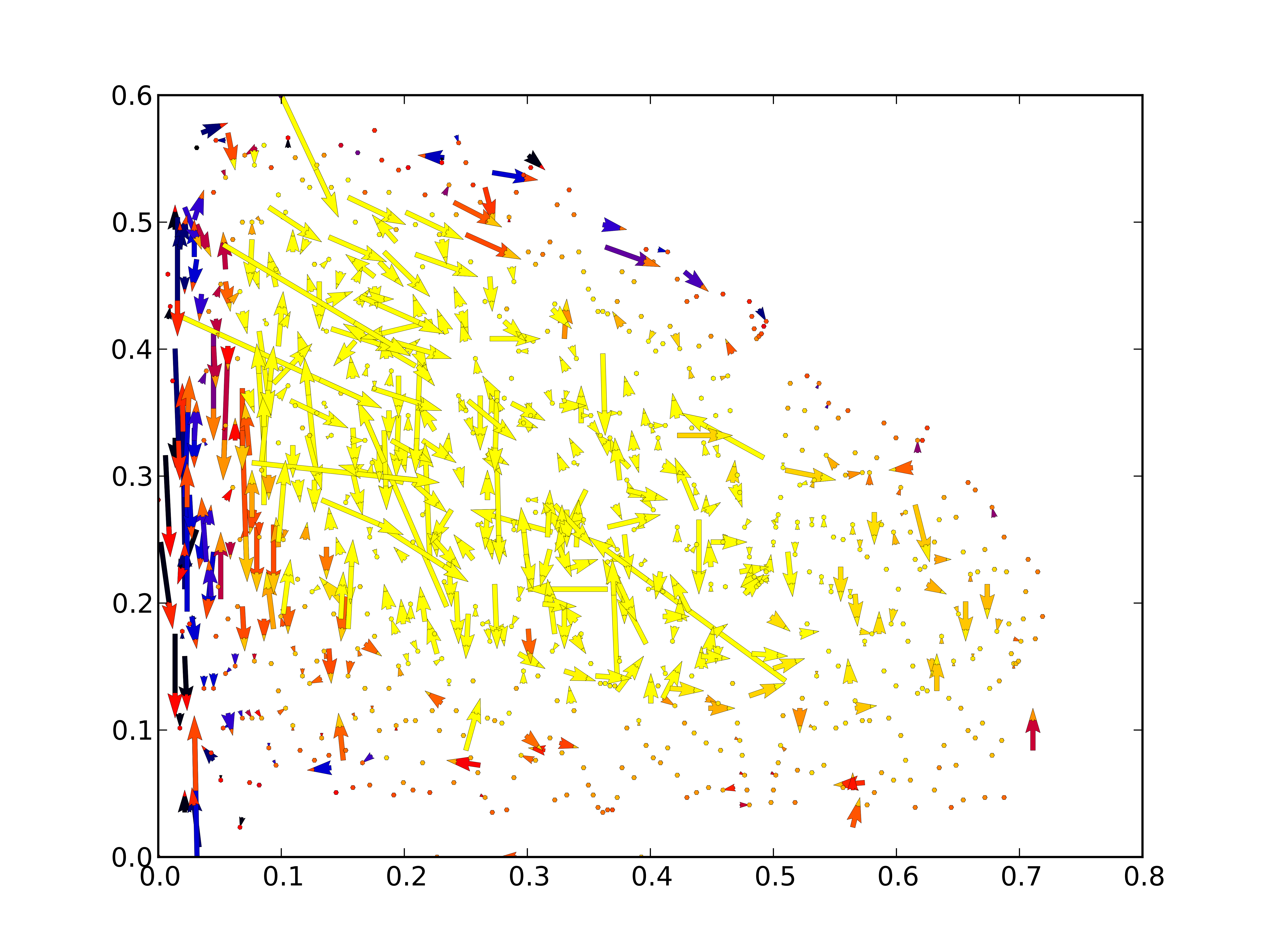
Serendipity

- We want our algorithms to capture serendipitous discoveries
- QD does that via Goal Switching

MAP-Elites



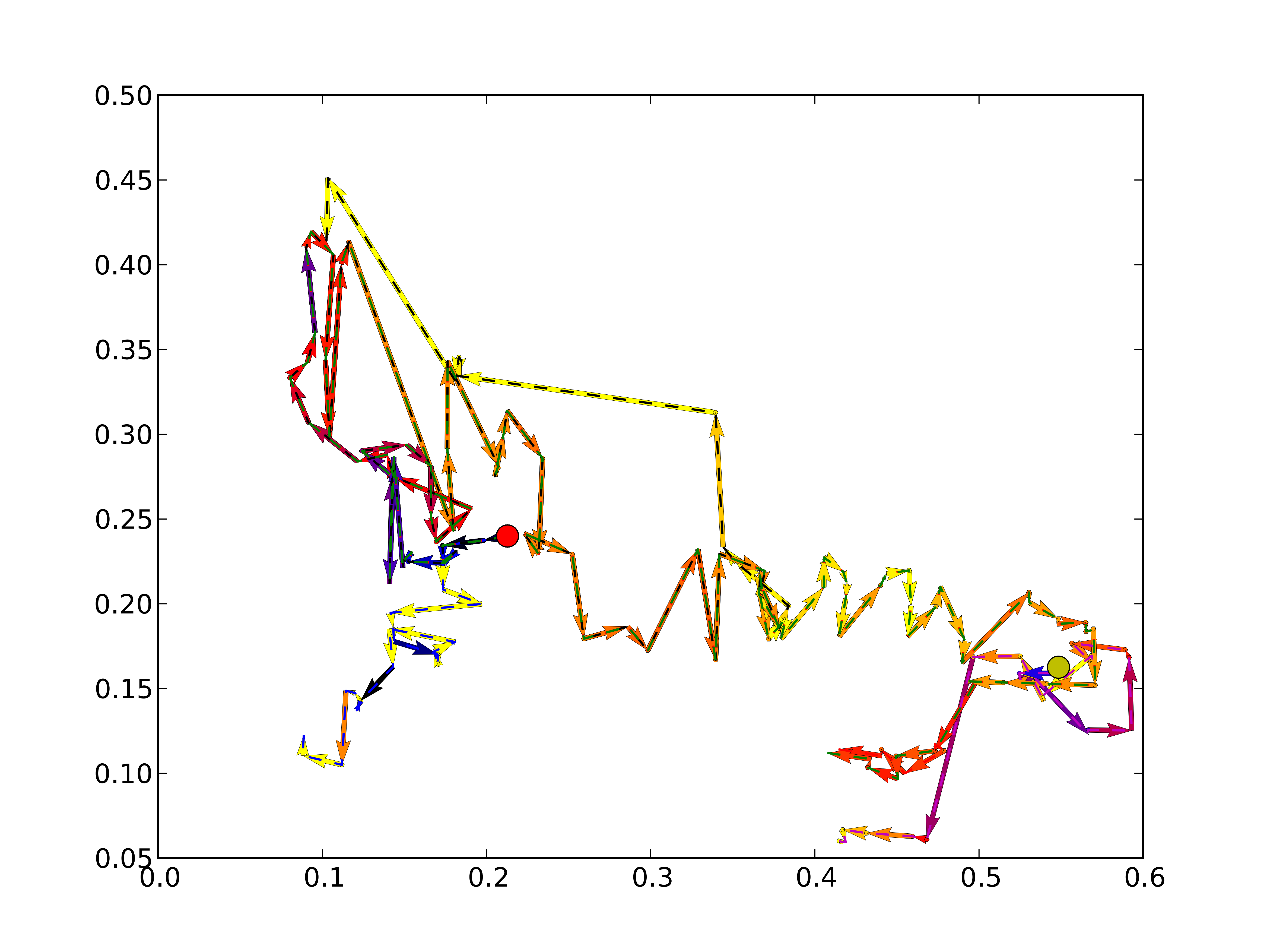
Goal Switching

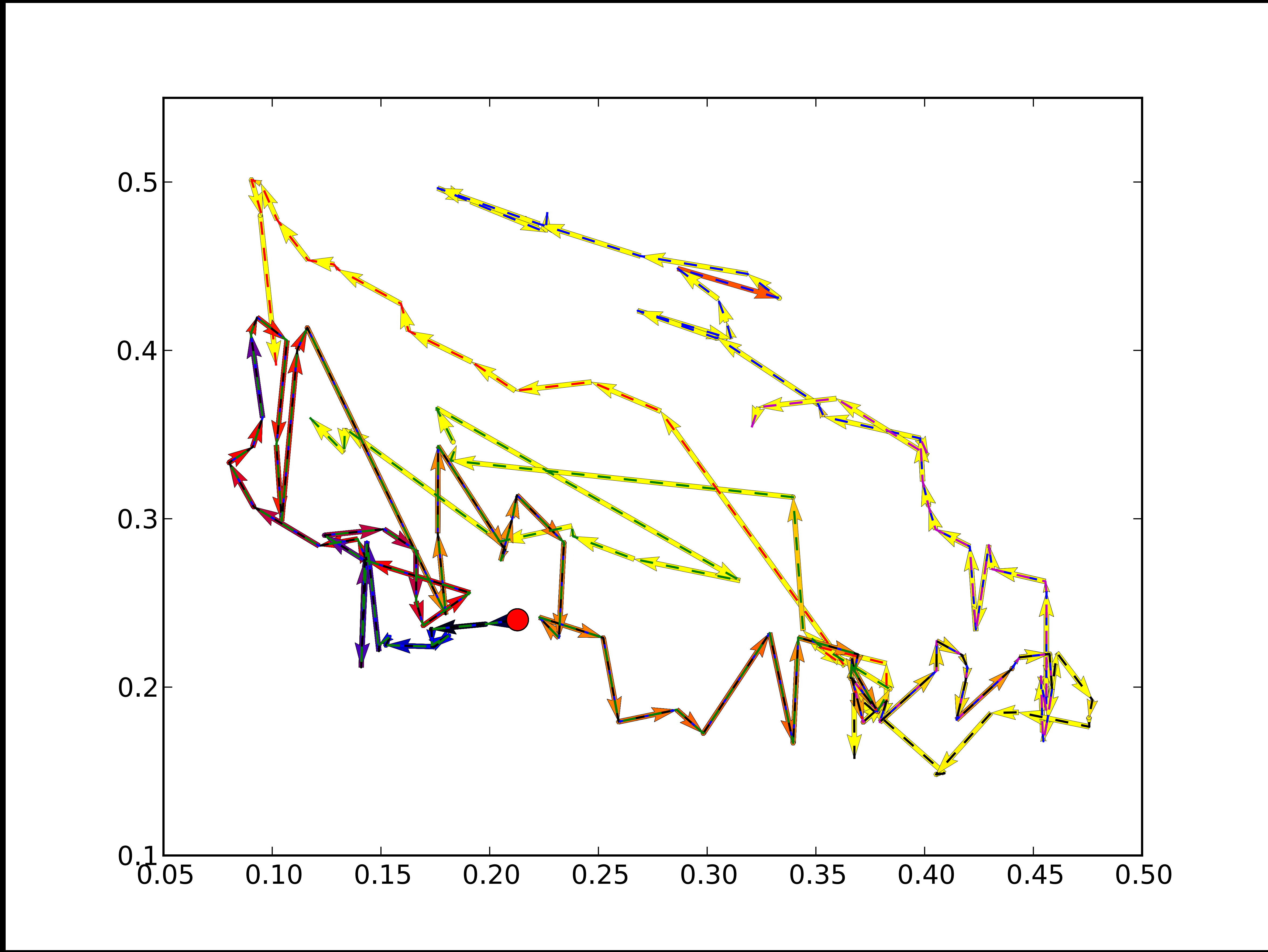


retina problem color = reward

MAP-Elites Mouret & Clune 2015

Automated Curricula Learning





MAP-Elites Lineages of a Few Final Solutions

Circles are iteration 0, color = reward

MAP-Elites Mouret & Clune 2015

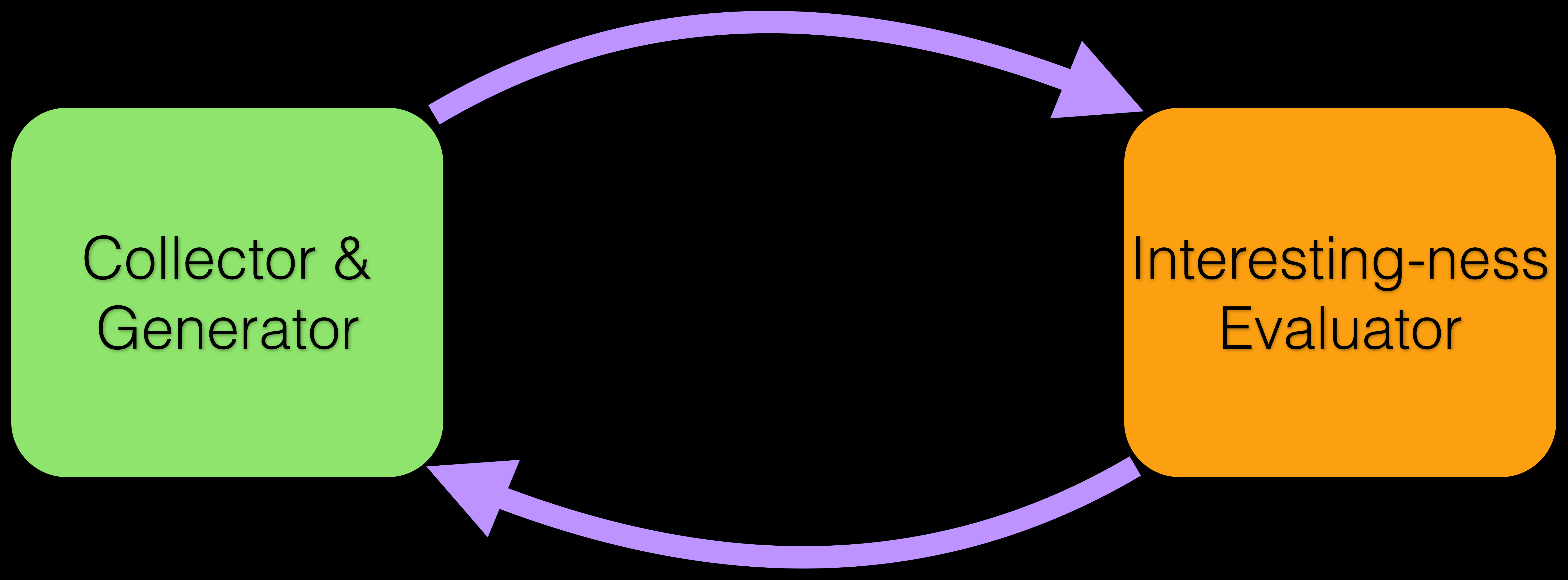
Innovation Engines

Nguyen, Yosinski & Clune 2015

- Nature, Culture, & QD algorithms are *Innovation Engines*
  - generate permutations of previous interesting things
  - if interesting, keep them
  - repeat

Innovation Engines

Nguyen, Yosinski & Clune 2015

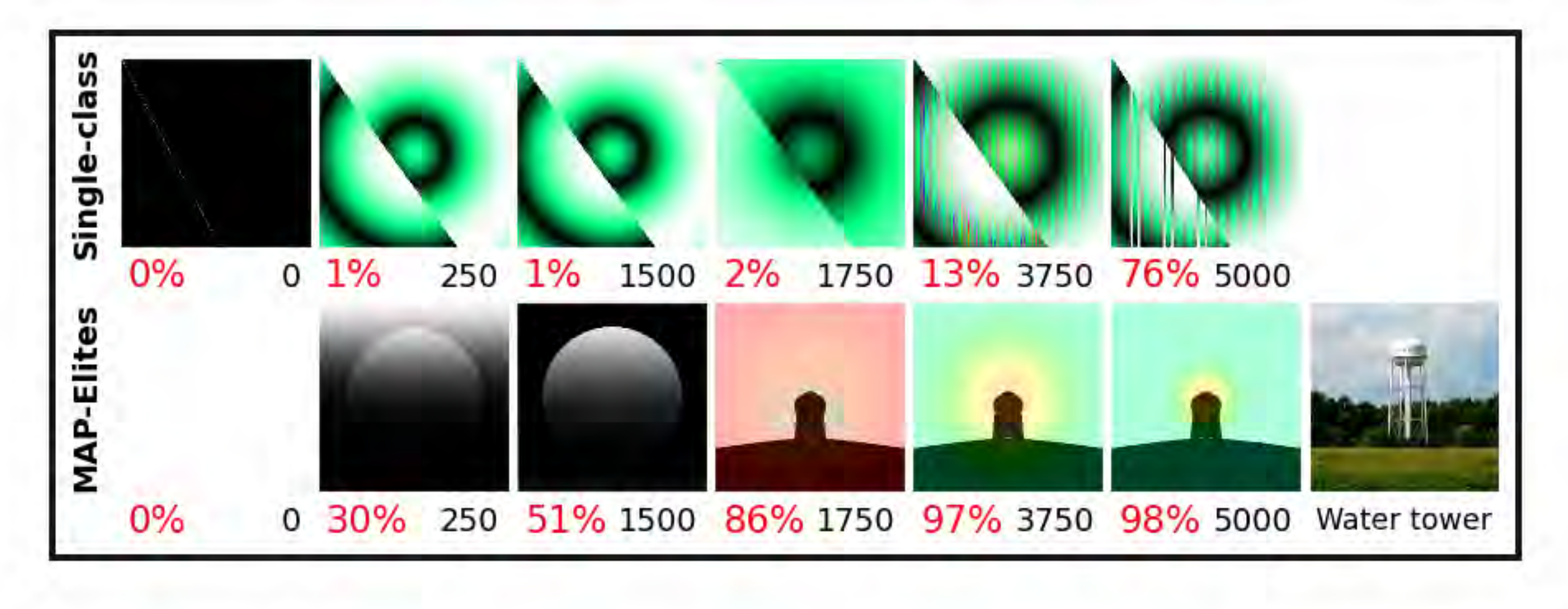


MAP-Elites
one bin per ImageNet class

AlexNet

Encodings: Small CPPN networks

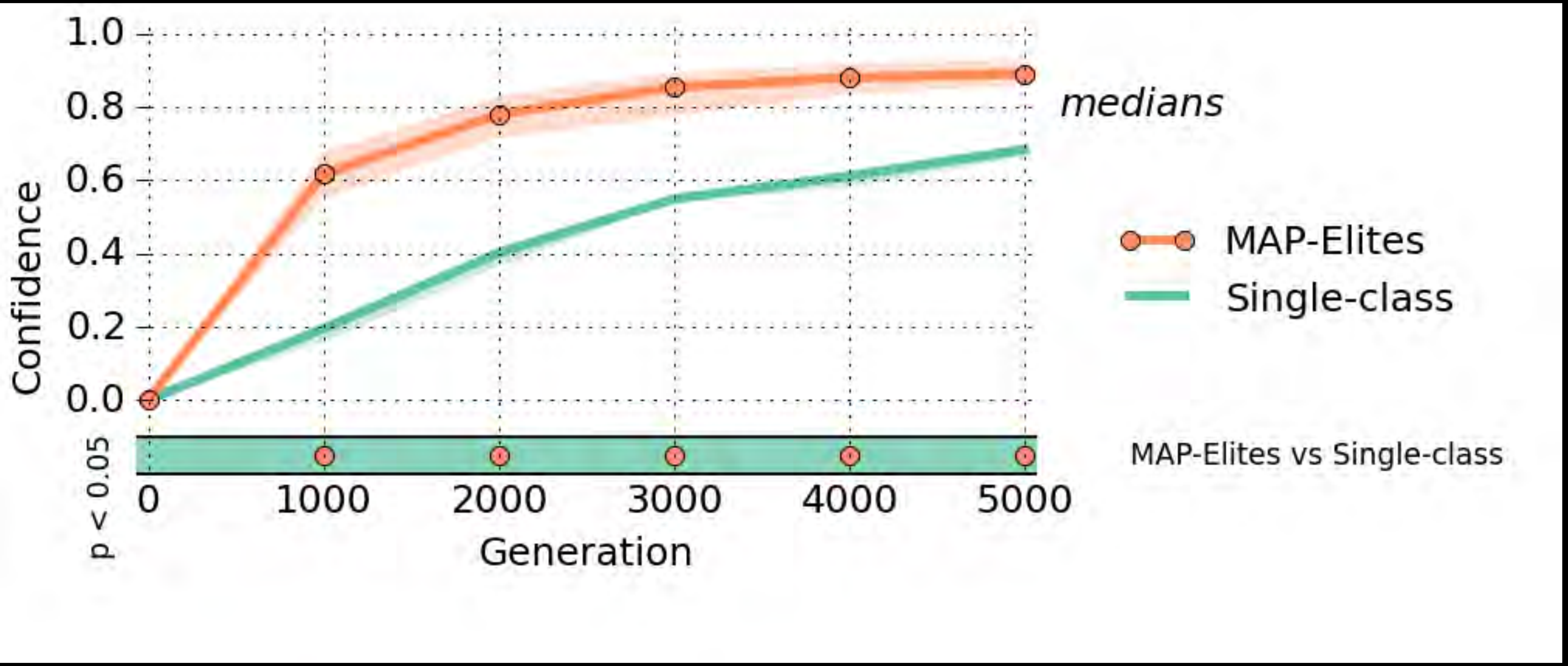
Goal Switching



Nguyen, Yosinski & Clune 2015

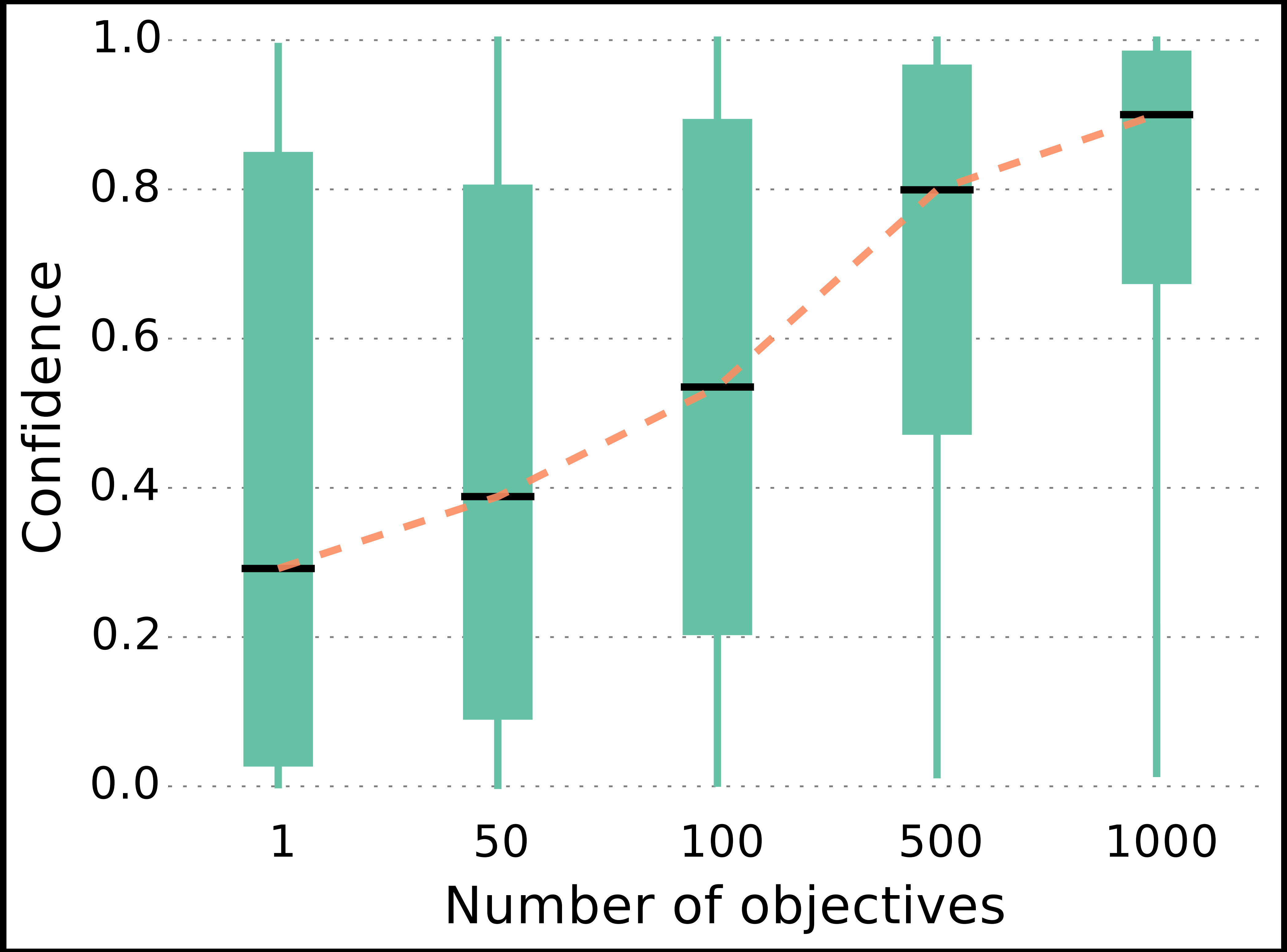
Goal Switching

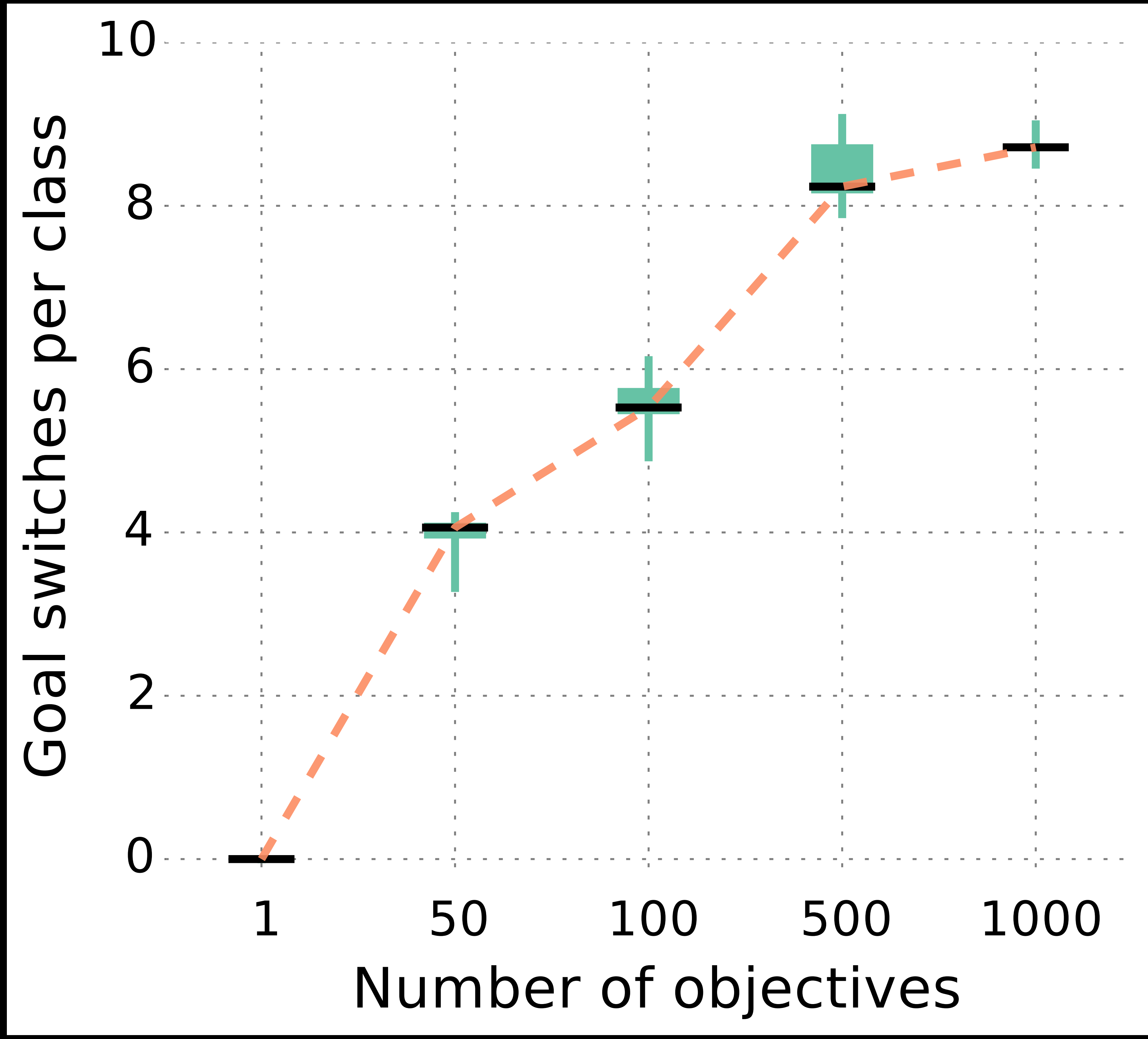
- Many-class MAP-Elites vs. One-class MAP-Elites



Nguyen, Yosinski & Clune 2015

Goal Switching

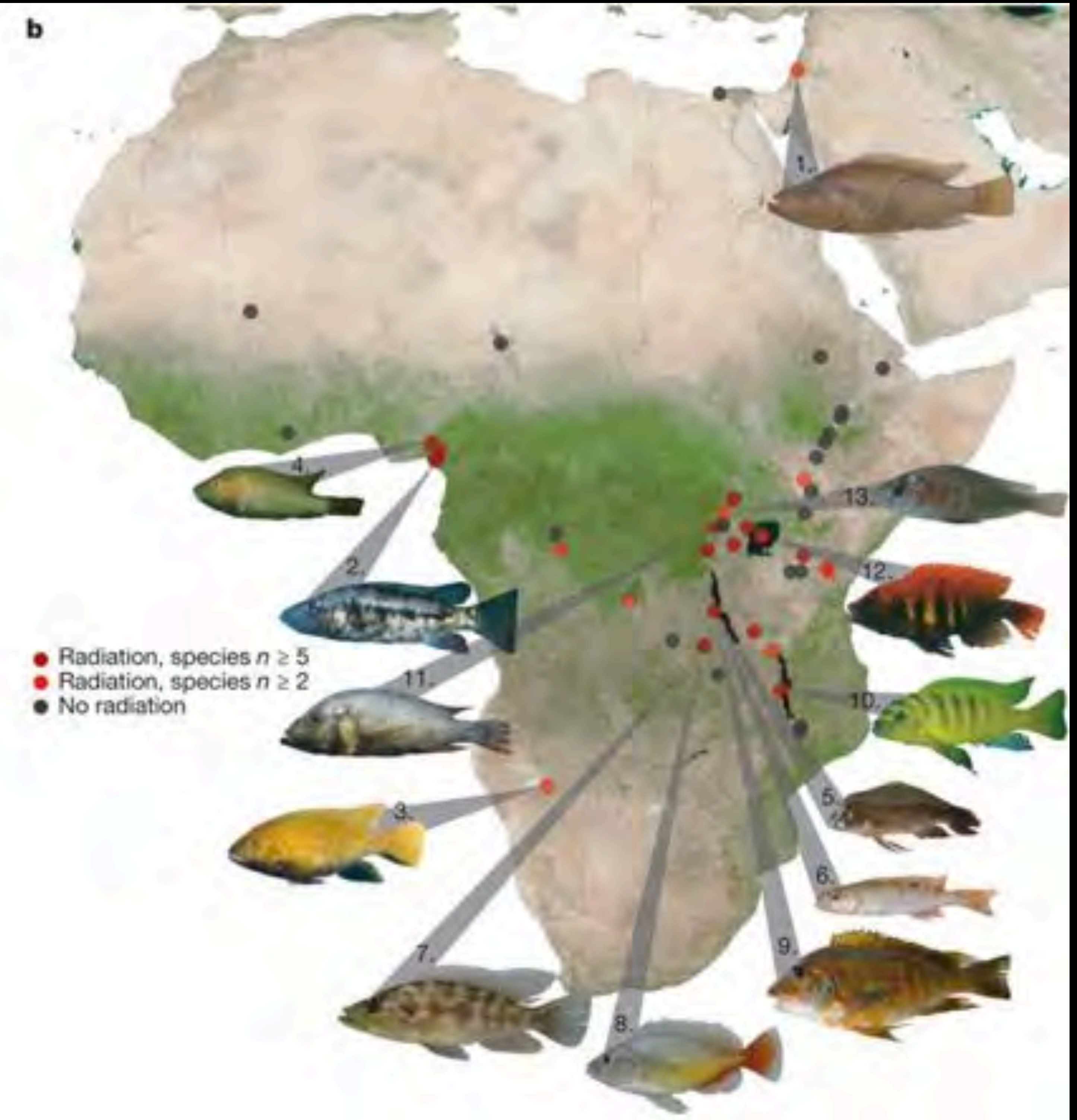


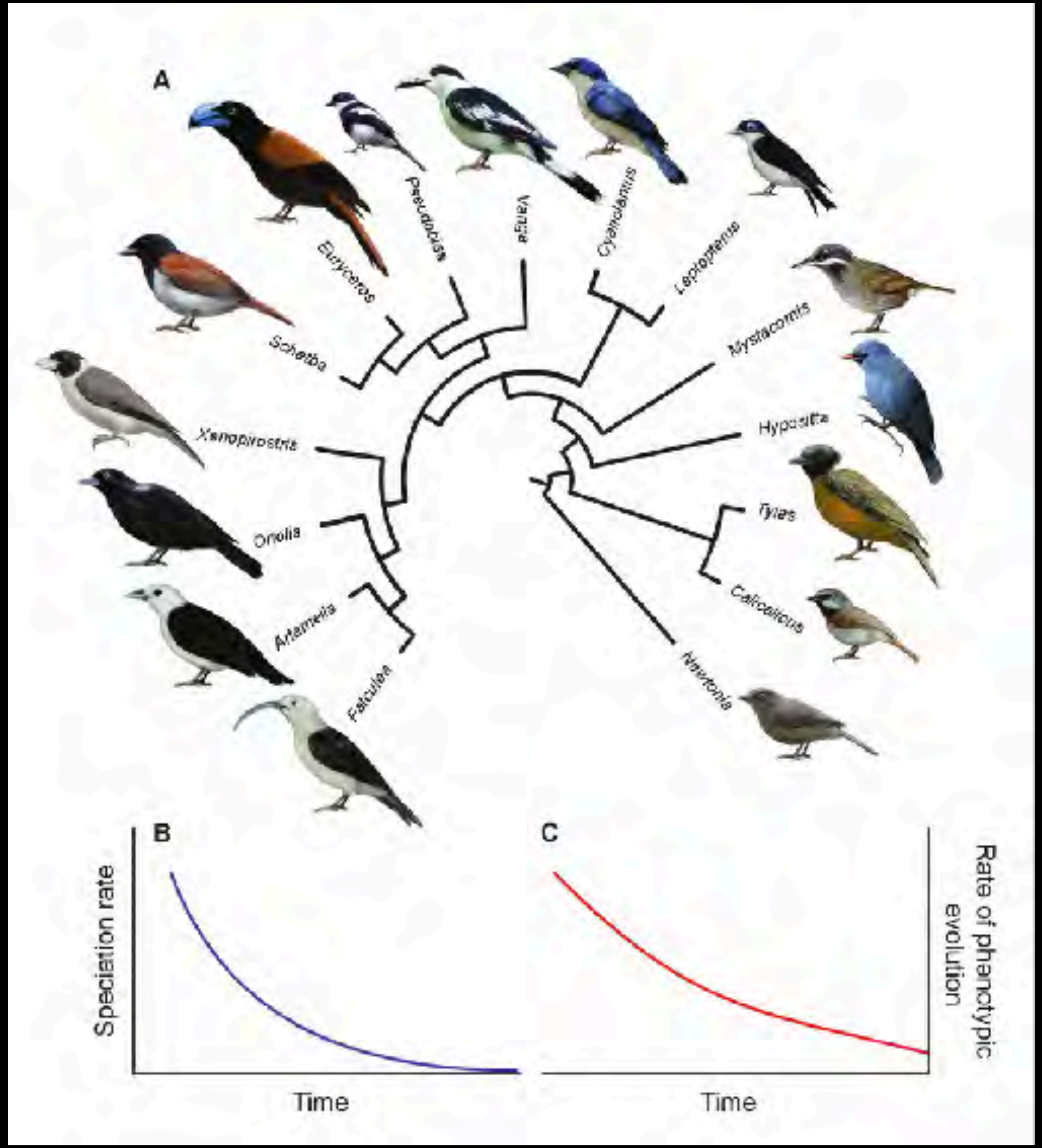


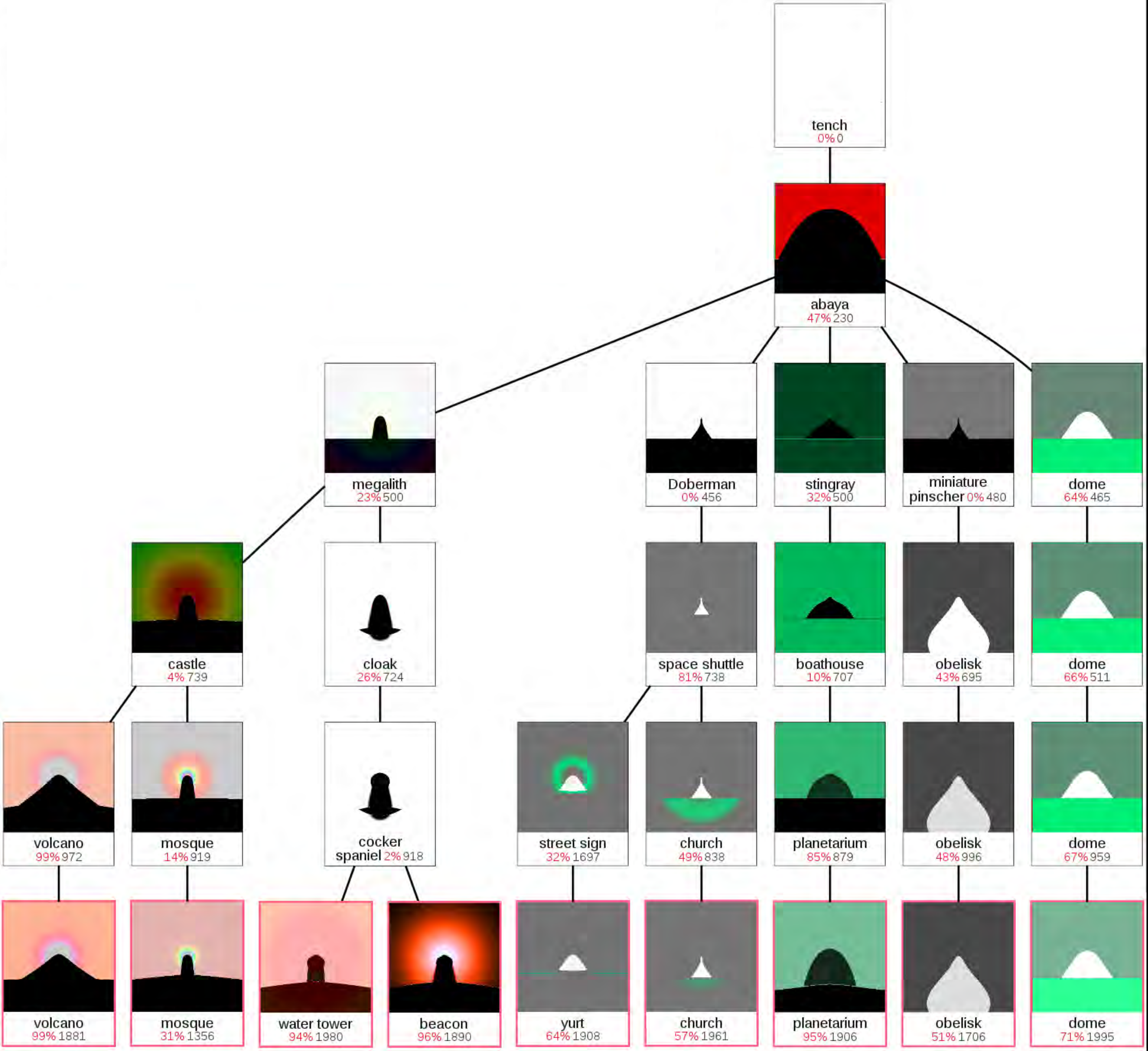
Nguyen, Yosinski & Clune 2015

Goal Switching Enables Good Ideas to Spread

- Fundamental advances spread to other problems/niches
- Then are built upon to solve that specific problem
- “Adaptive Radiations”







Adaptive Radiations in QD!

Innovation Engines

Nguyen, Yosinski & Clune 2015

Hindsight Experience Replay

Andrychowicz et al. 2017

- RL algorithm
  - single agent
  - uses goal-conditioned Q-learning
- Try to go to a goal
- If you end up somewhere else, pretend that was your goal
  - goal switching!
- Eventually learn the highest-quality way to do a diverse set of things
  - effectively is a QD algorithm
    - where the “population” is in goals for one agent, not a population of agents

Multi-Modal Agents

CMOEA. Huizinga & Clune 2018

- Wanted: robots that can perform many different actions/skills
  - in different contexts (e.g. options hierarchical RL)
  - solve different problems
- Insight: QD algorithms can help produce such generalists

Move Forward

Move Backward

Turn Left

Turn Right

Jump

Multi-Modal Agents

CMOEA. Huizinga & Clune 2018

- A curriculum probably helps
- Which one?

Move Forward

Move Backward

Turn Left

Turn Right

Jump

CMOEA

Huizinga & Clune 2018

- Idea: one niche per
  - single task
  - combination of tasks

CMOEA

Huizinga & Clune 2018

All Tasks

...

Move Forward, Move Backward Turn Left

Move Forward, Move Backward Turn Right

Move Forward, Move Backward Jump

...

Move Backward Turn Left Jump

Move Backward Turn Right Jump

Move Forward Move Backward

Move Forward Turn Left

Move Forward Turn Right

Move Forward Jump

...

Turn Right & Jump

Move Forward

Move Backward

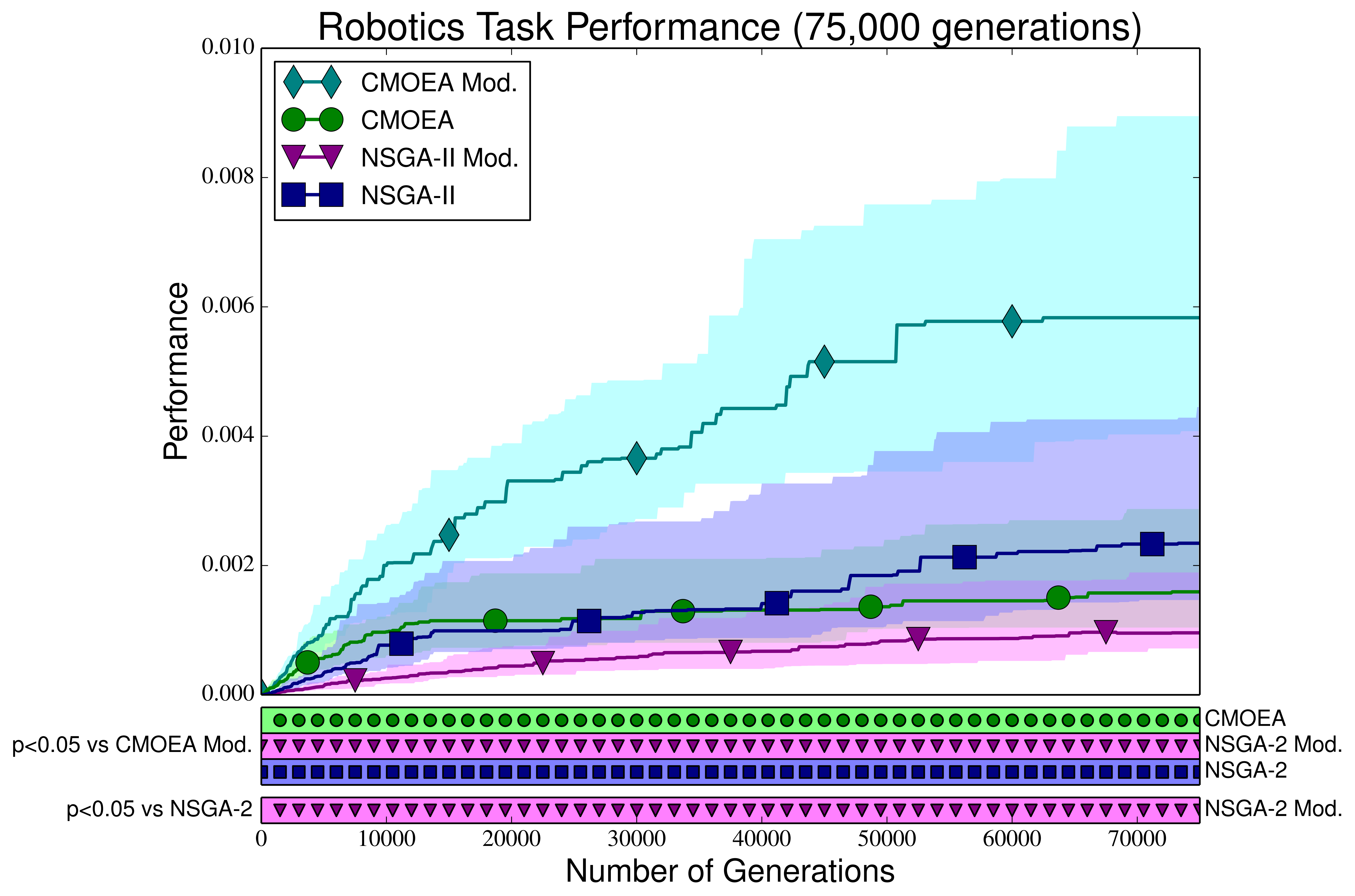
Turn Left

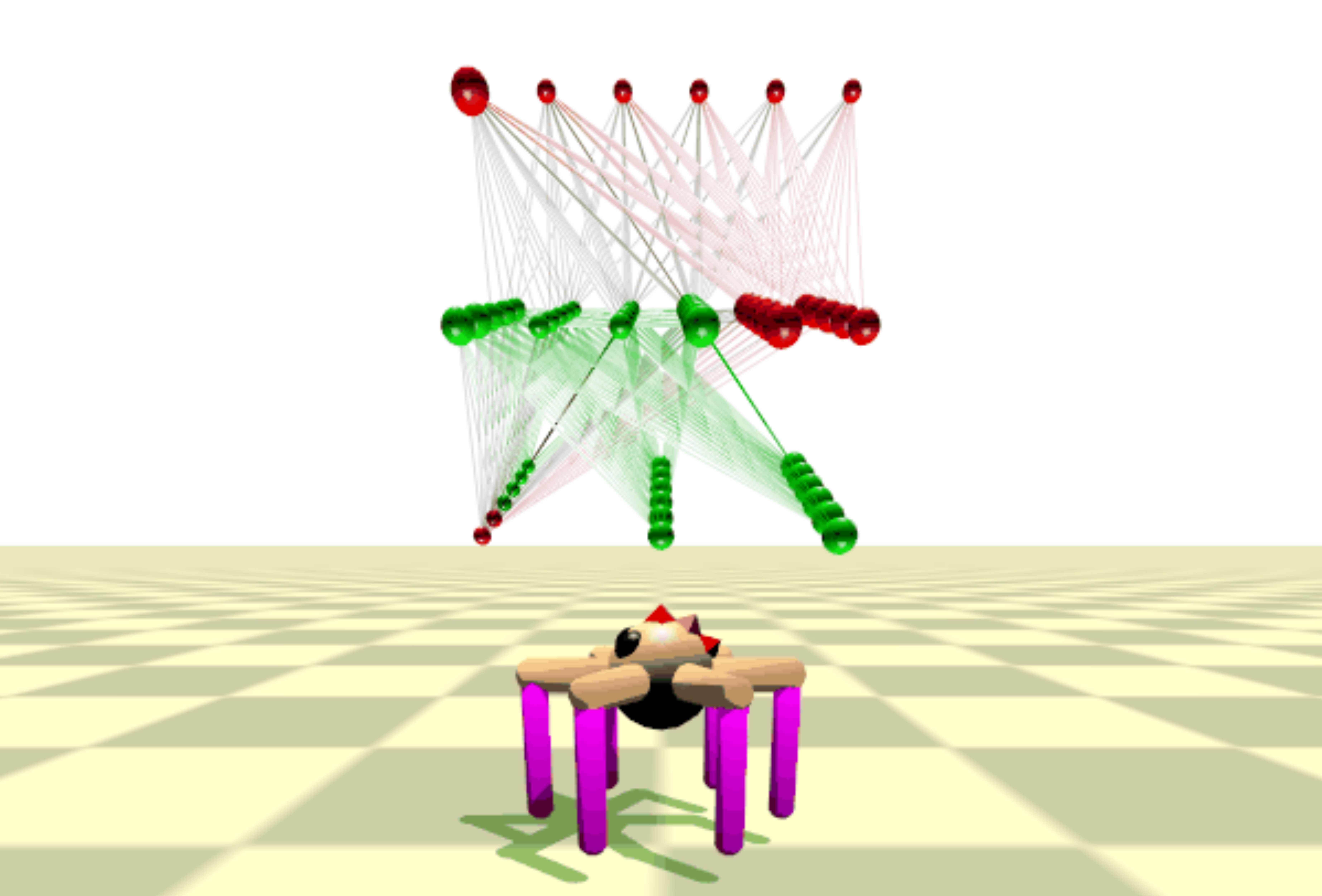
Turn Right

Jump

CMOEA

Huizinga & Clune 2018





Move forward

Other Applications of Quality Diversity Algorithms

Robots that adapt like animals

Nature 2015

Antoine Cully
UPMC Université
France

Jeff Clune
University of Wyoming
USA

Danesh Tarapore
UPMC Université
France

Jean-Baptiste Mouret
UPMC Université
France


iRobot

Damage Recovery

Modern, Learning-Based Approaches

- Simple robots (low-dimensional state & action spaces)
- Require lots of real-world trials

Yosinski et al. 2013

Kohl & Stone 2004

Bongard et al. 2006

Animals

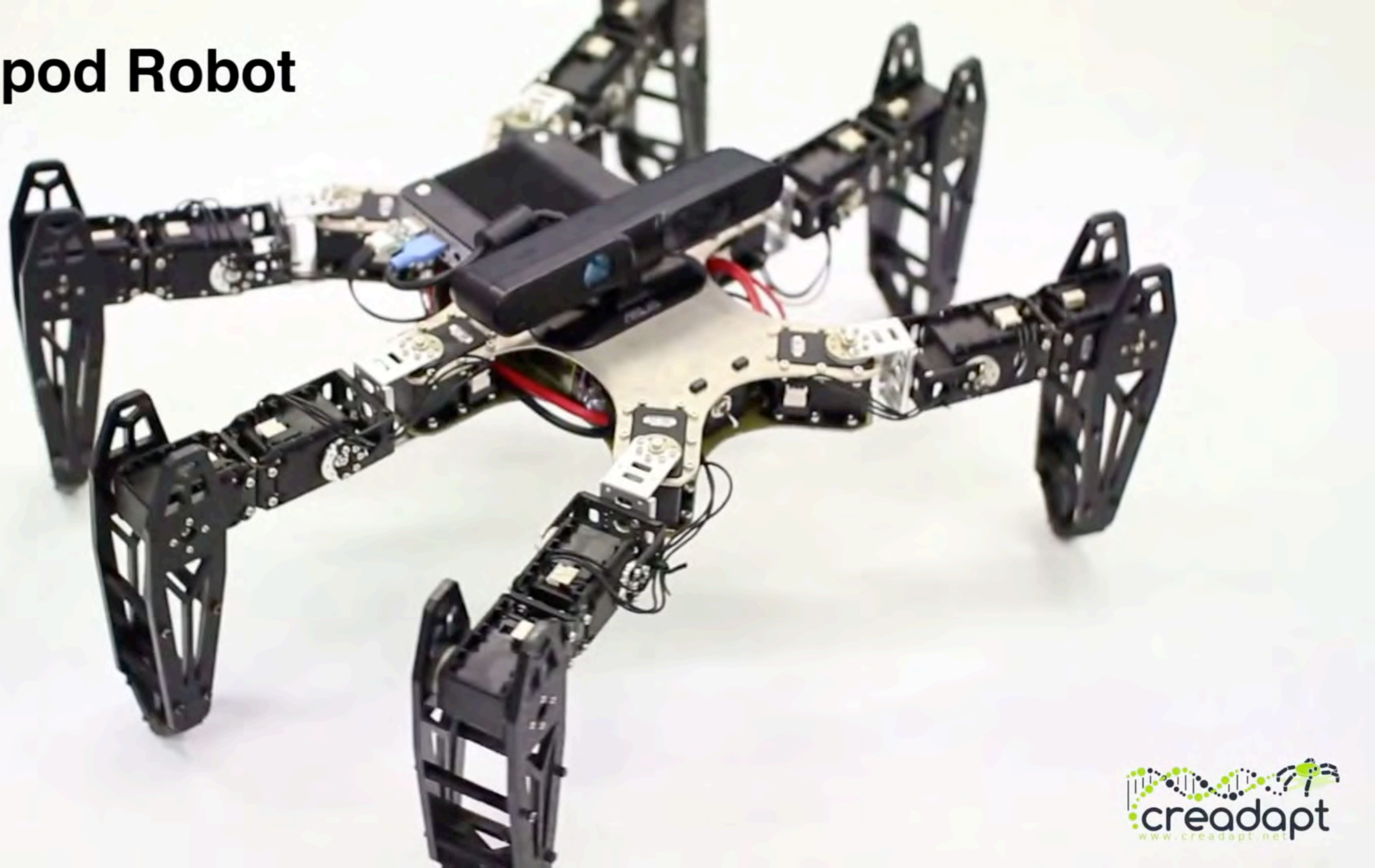
- Have intuitions about different ways to move
- Conduct a few, intelligent tests
- Pick a behavior that works despite injury

Robots that Adapt Like Animals

- Have intuitions about different ways to move
- Conduct a few, intelligent tests
- Pick a behavior that works despite injury

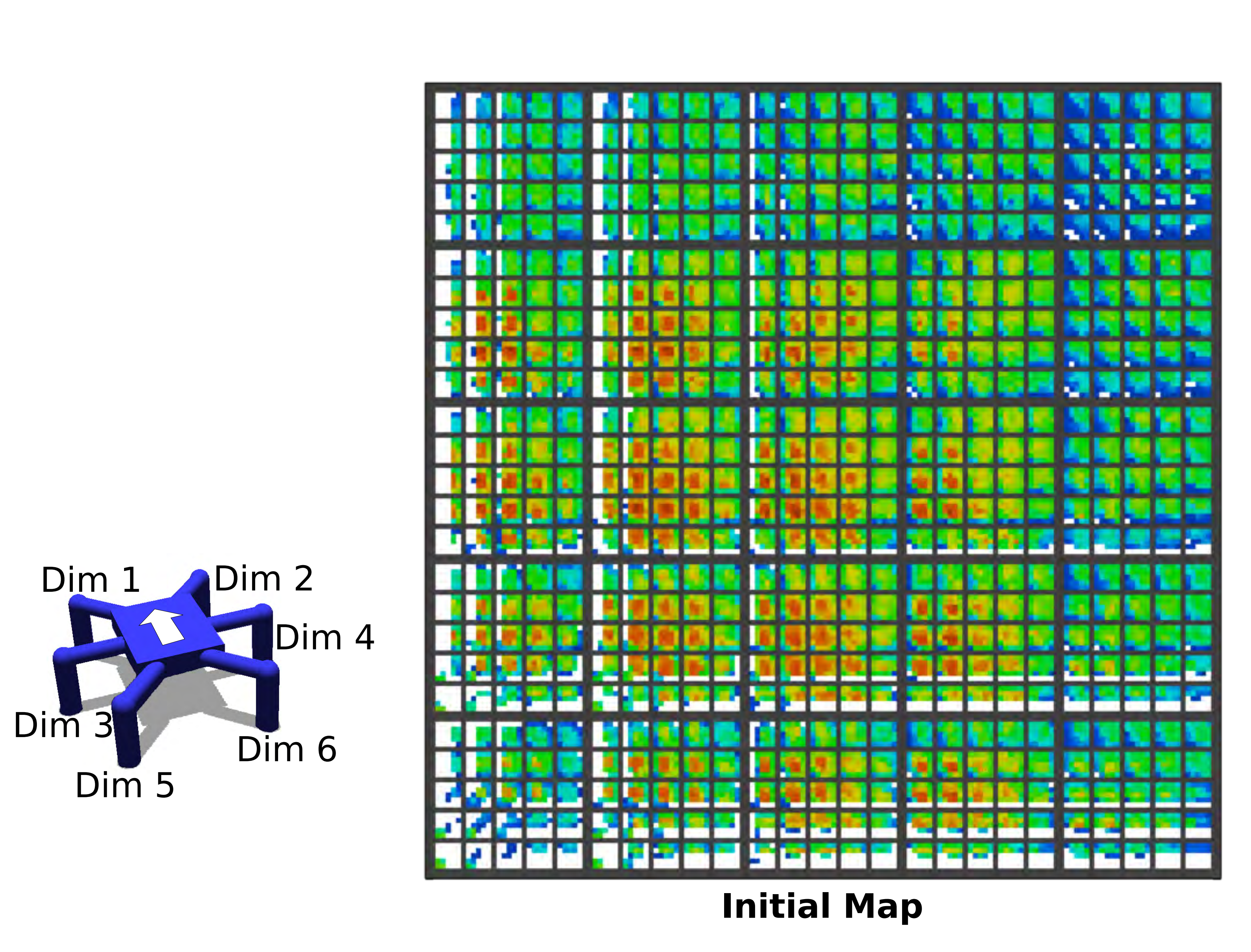
intuitions about different ways to move → few, intelligent tests → pick one that works despite injury

Hexapod Robot

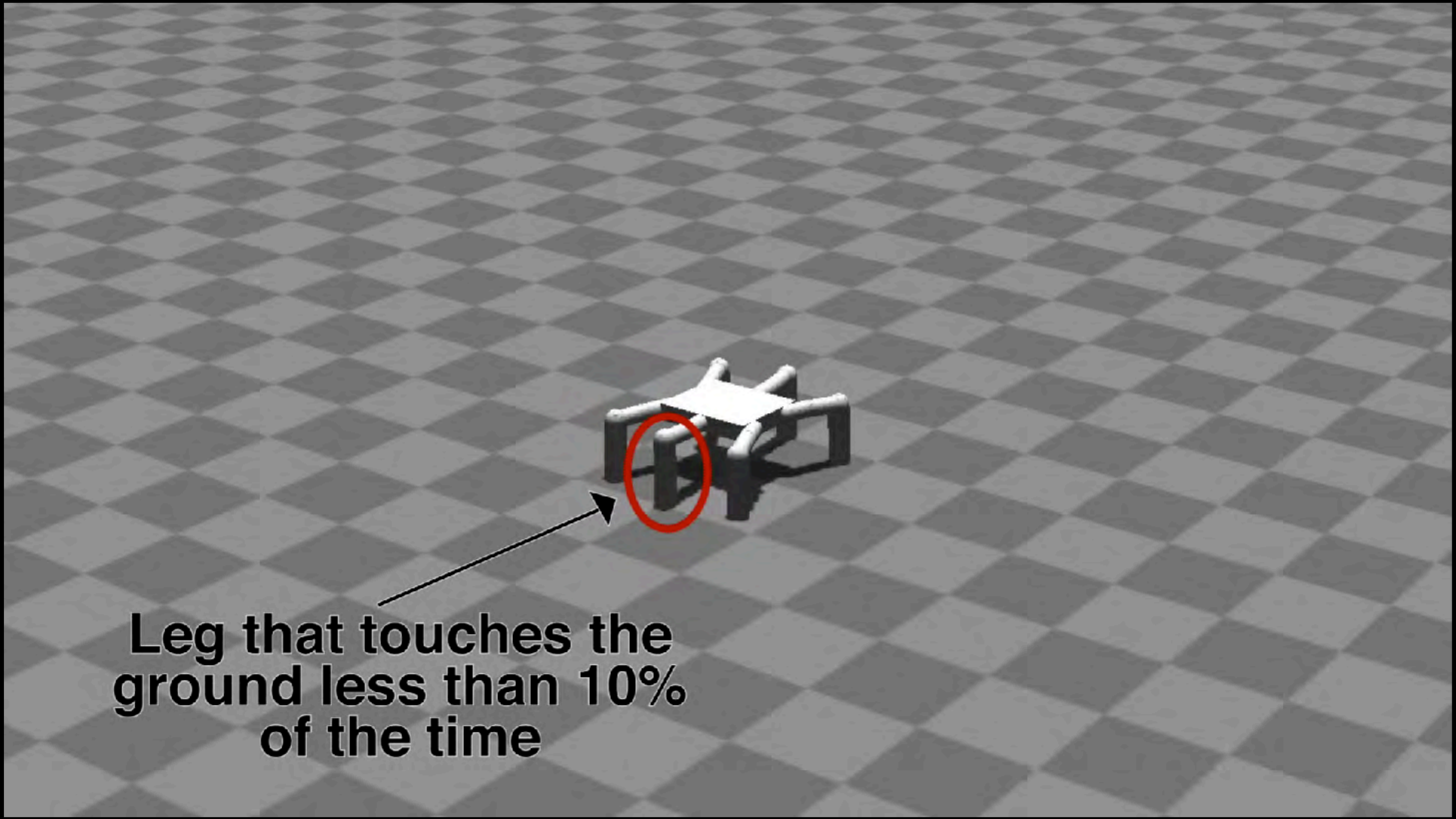


intuitions about
different ways to move

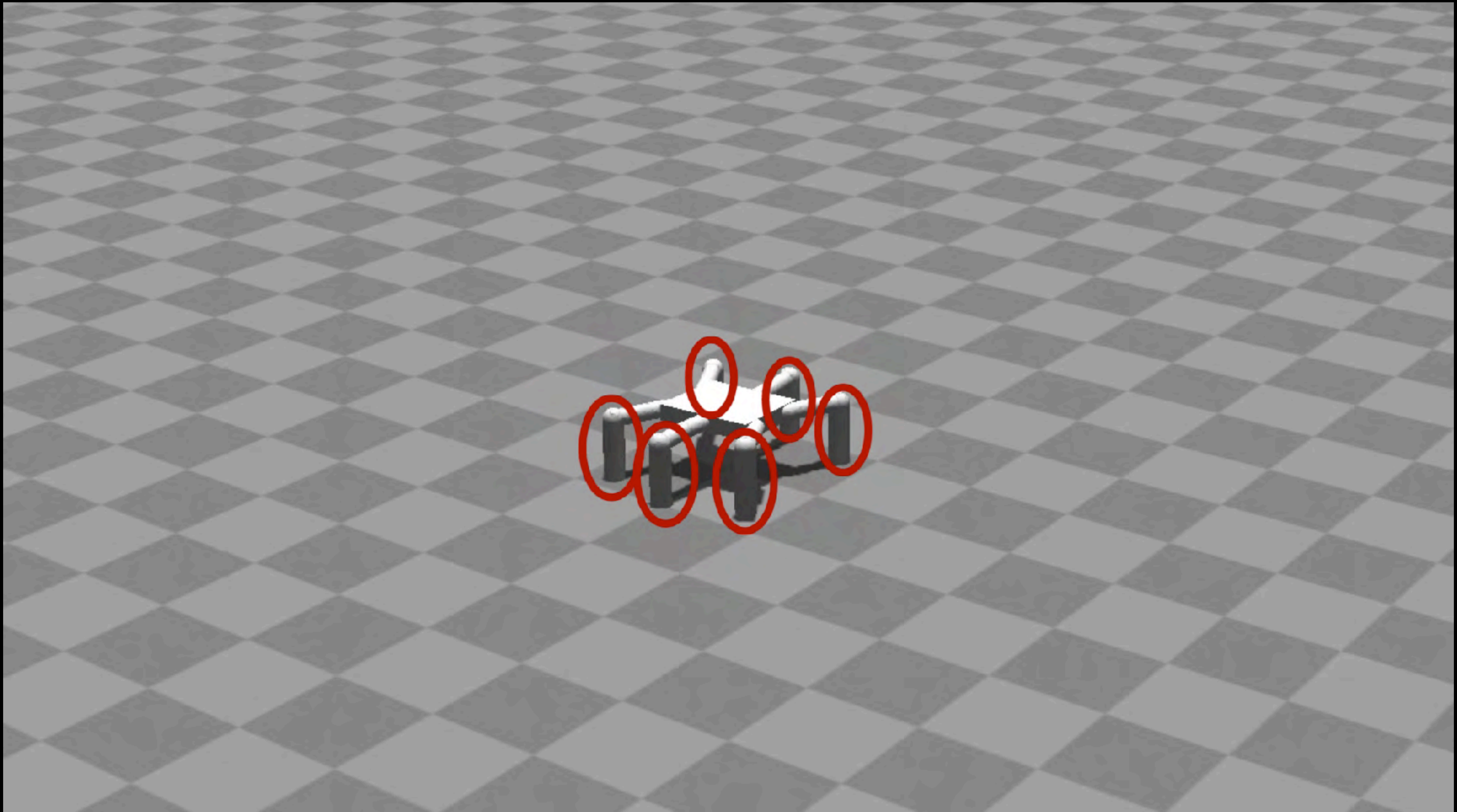
- MAP-Elites
- Behavioral characterization
  - % of time each leg touches the ground (6-dimensional)
- Massive search space
- MAP-Elites map has ~13,000 diverse, high-performing gaits



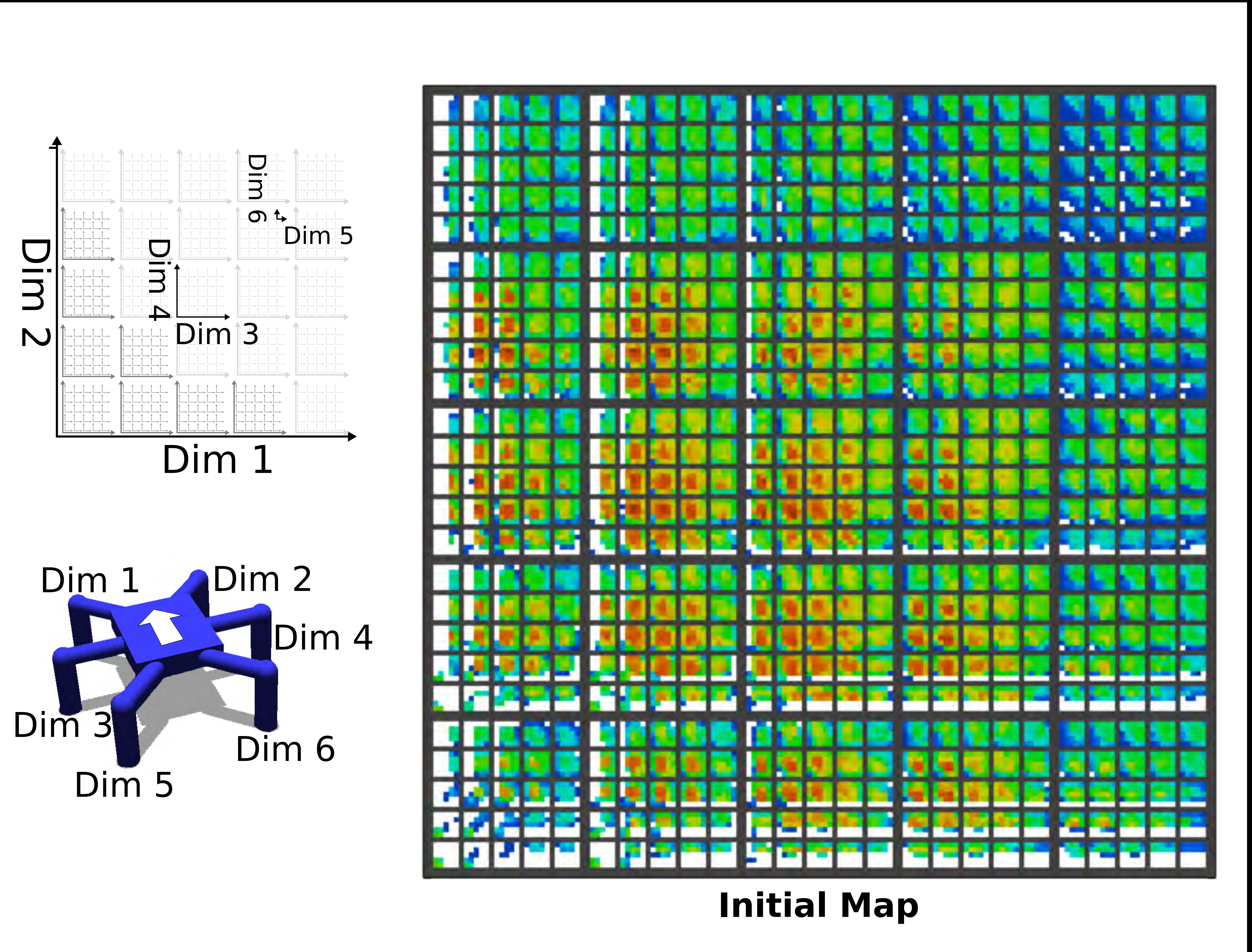
intuitions about
different ways to move



Corner Case: Feet never touch the ground

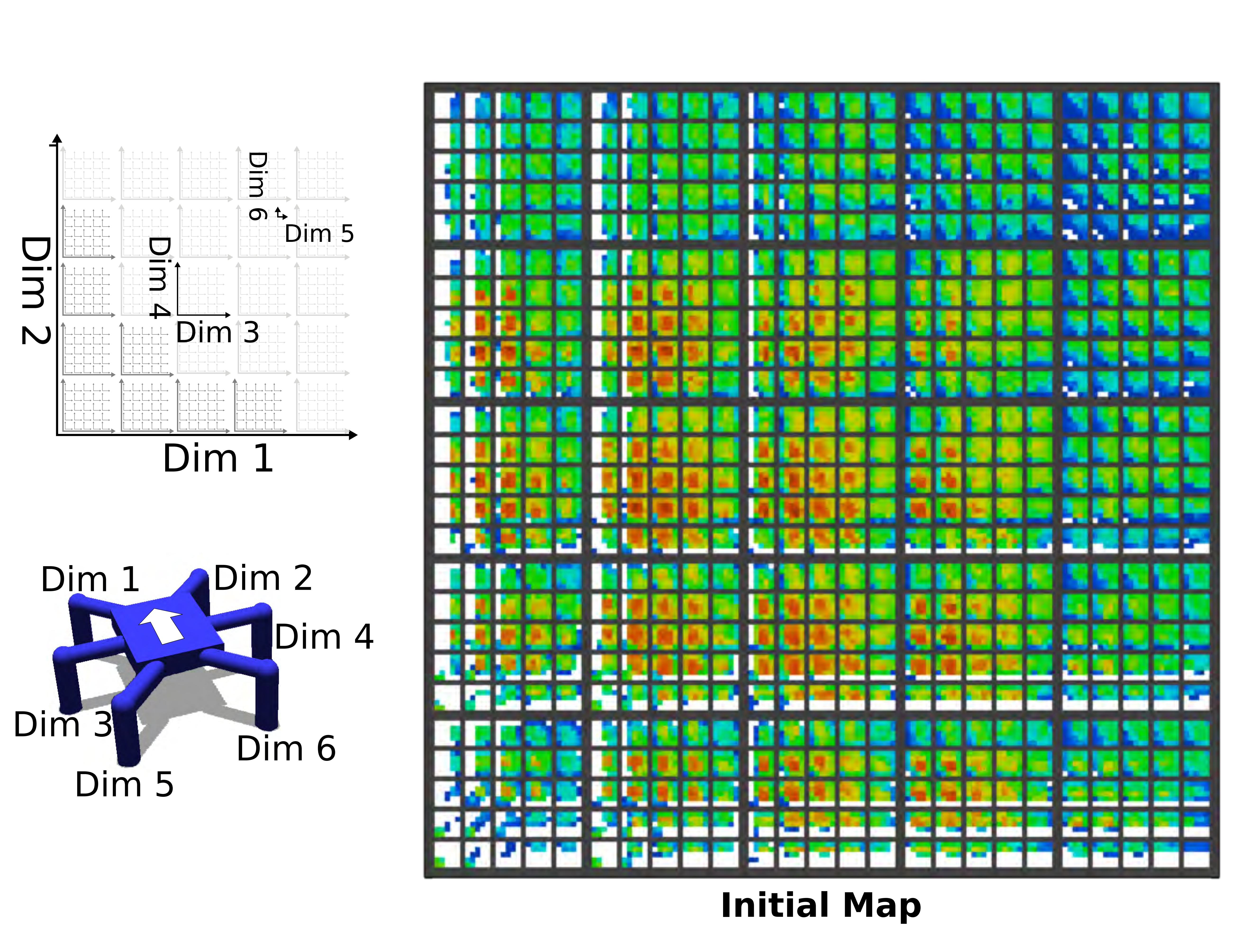


intuitions about different ways to move



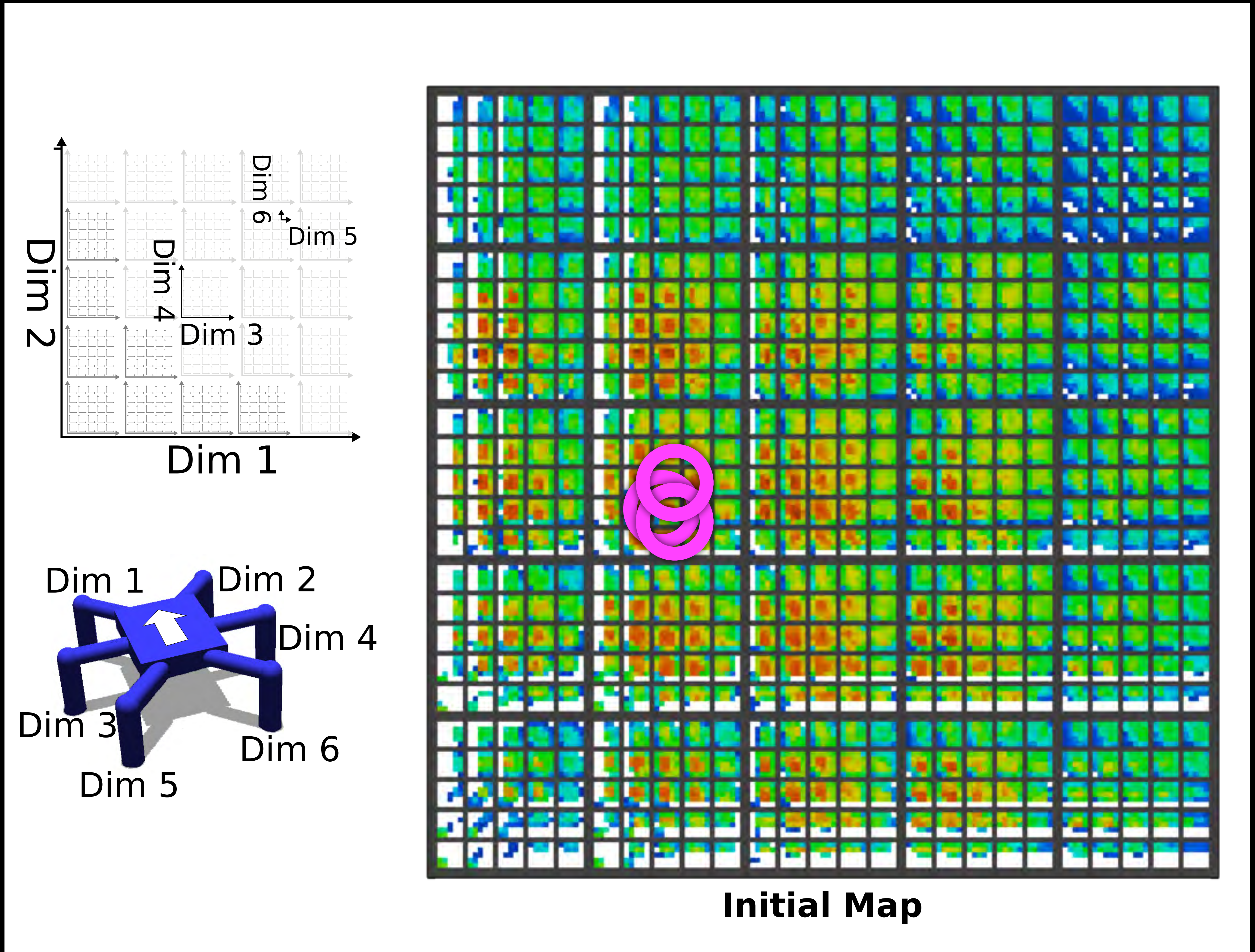
On the simulated, undamaged robot

intuitions about different ways to move → few, intelligent tests



Which behaviors should we test?

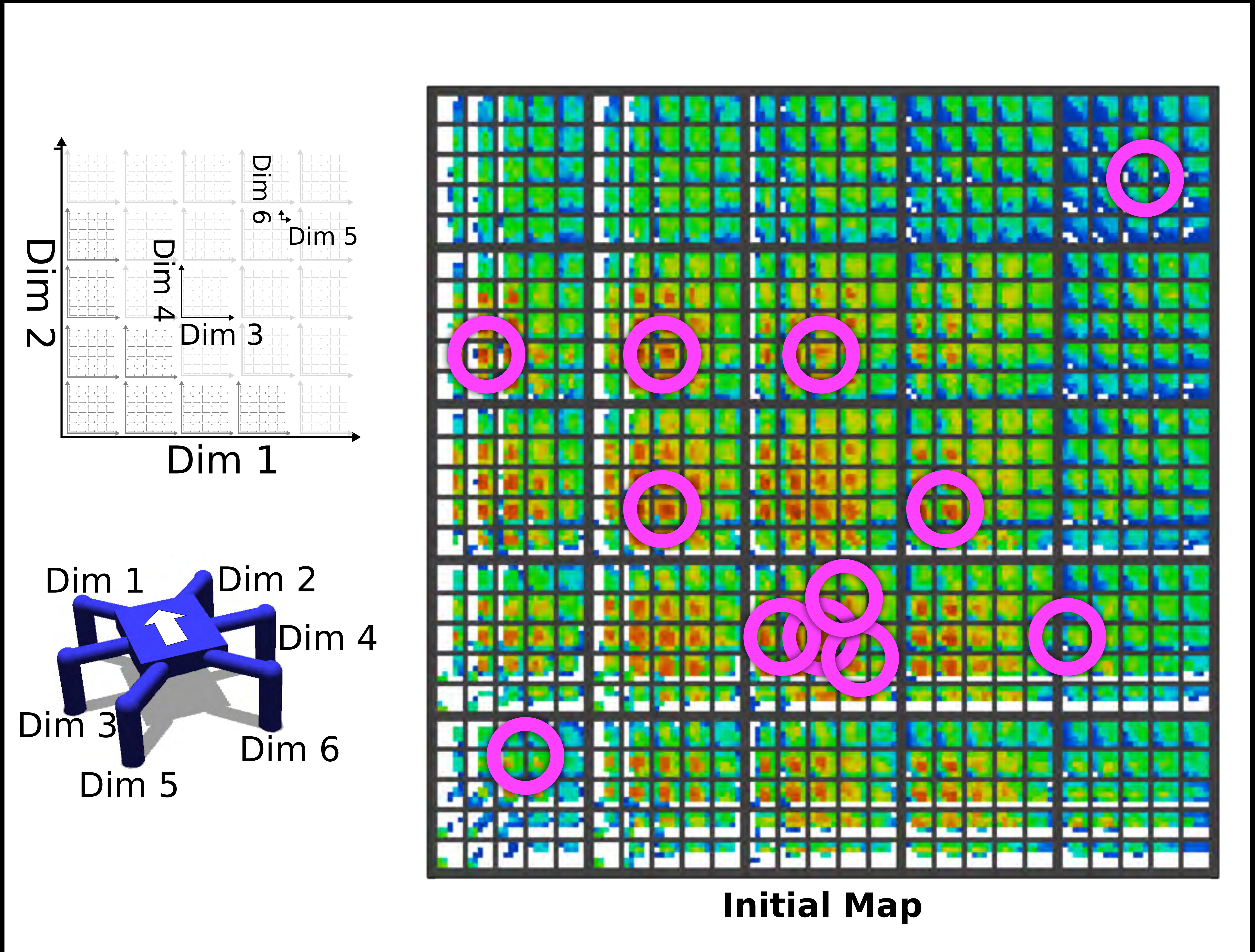
intuitions about different ways to move → few, intelligent tests



Could try top N:

But they are likely very similar.

intuitions about different ways to move → few, intelligent tests



Bayesian Optimization:

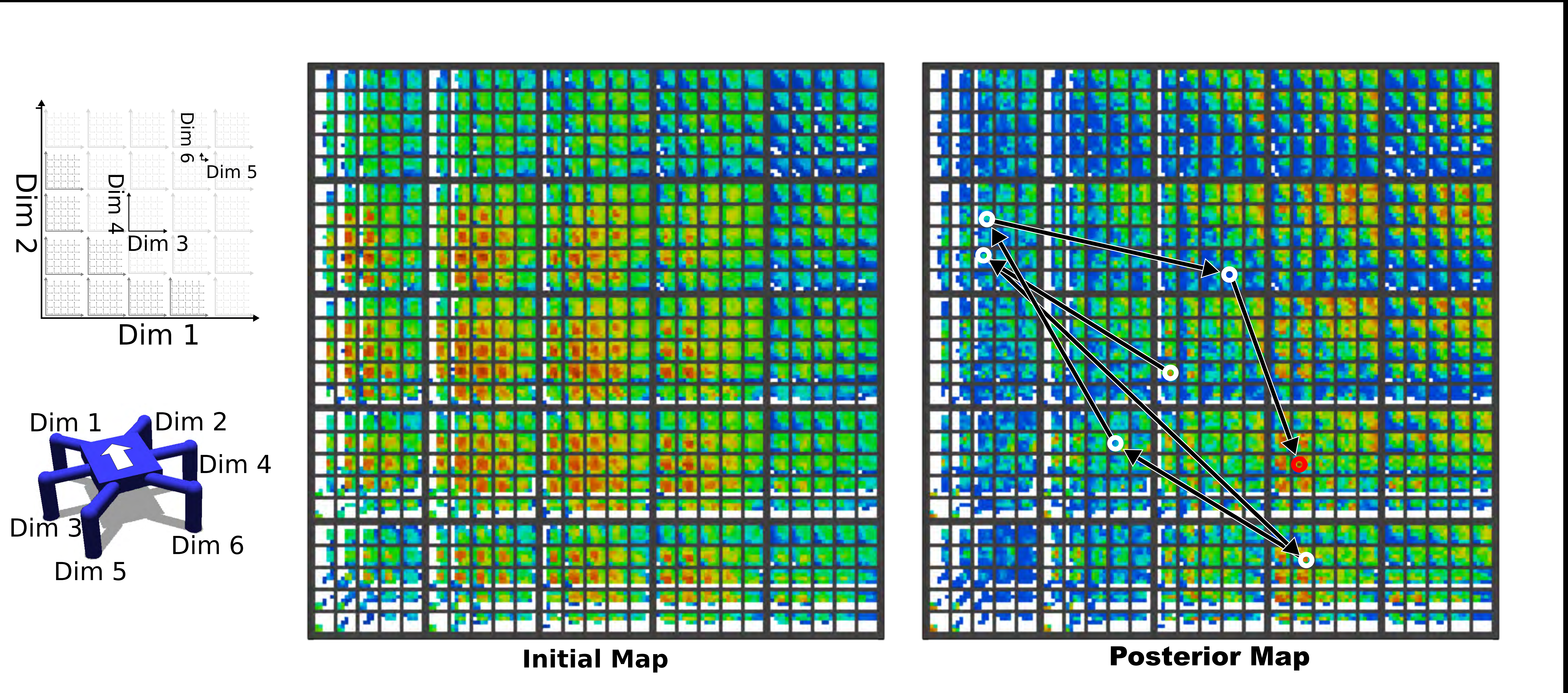
Tries different types solutions

Bayesian Optimization

Prior:
MAP-Elites Map

Posterior:
Map updated after real-world tests

Stop when:
A real-world behavior is >90% of best untested point



One-dimensional Example

“Intelligent Trial & Error”

intuitions about different ways to move → few, intelligent tests → pick one that works despite injury

MAP-Elites Map

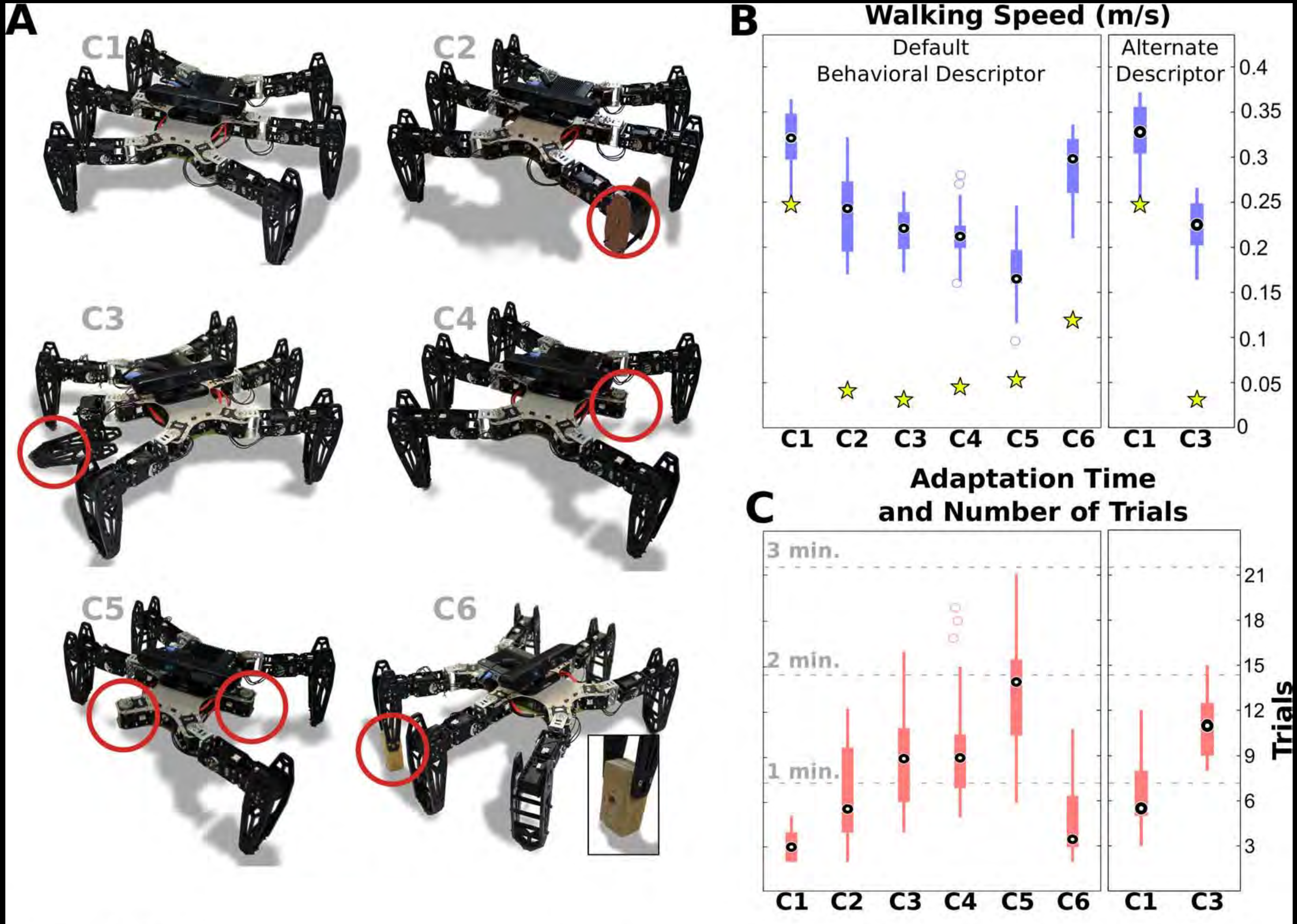
Bayesian Optimization w Map as Prior

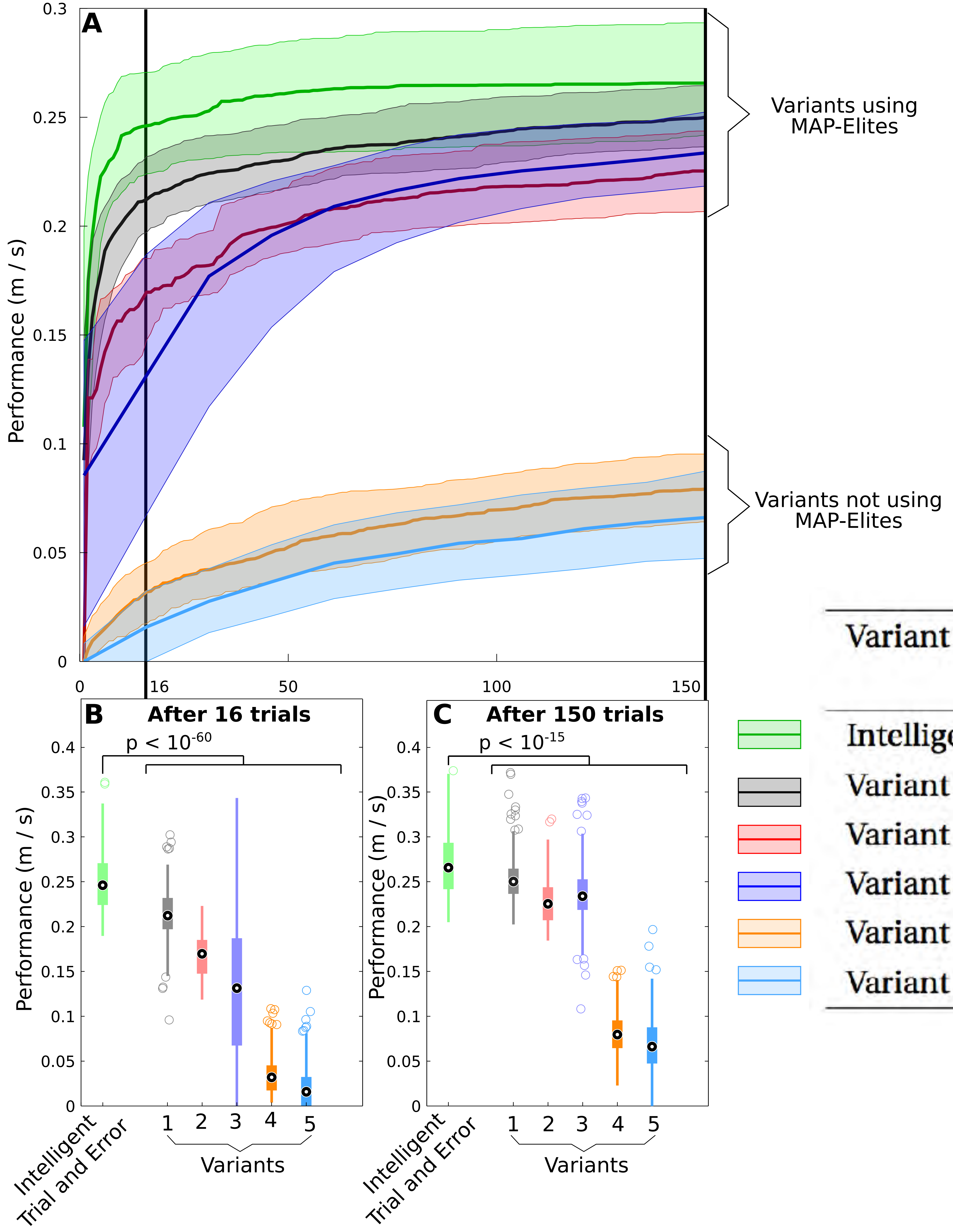
Found >90% of Best Possible

Undamaged robot controlled with classic tripod gait


00:00:00
Behavior-performance Map
Forward Speed (m/s)
0.13
Trajectory
GUIL

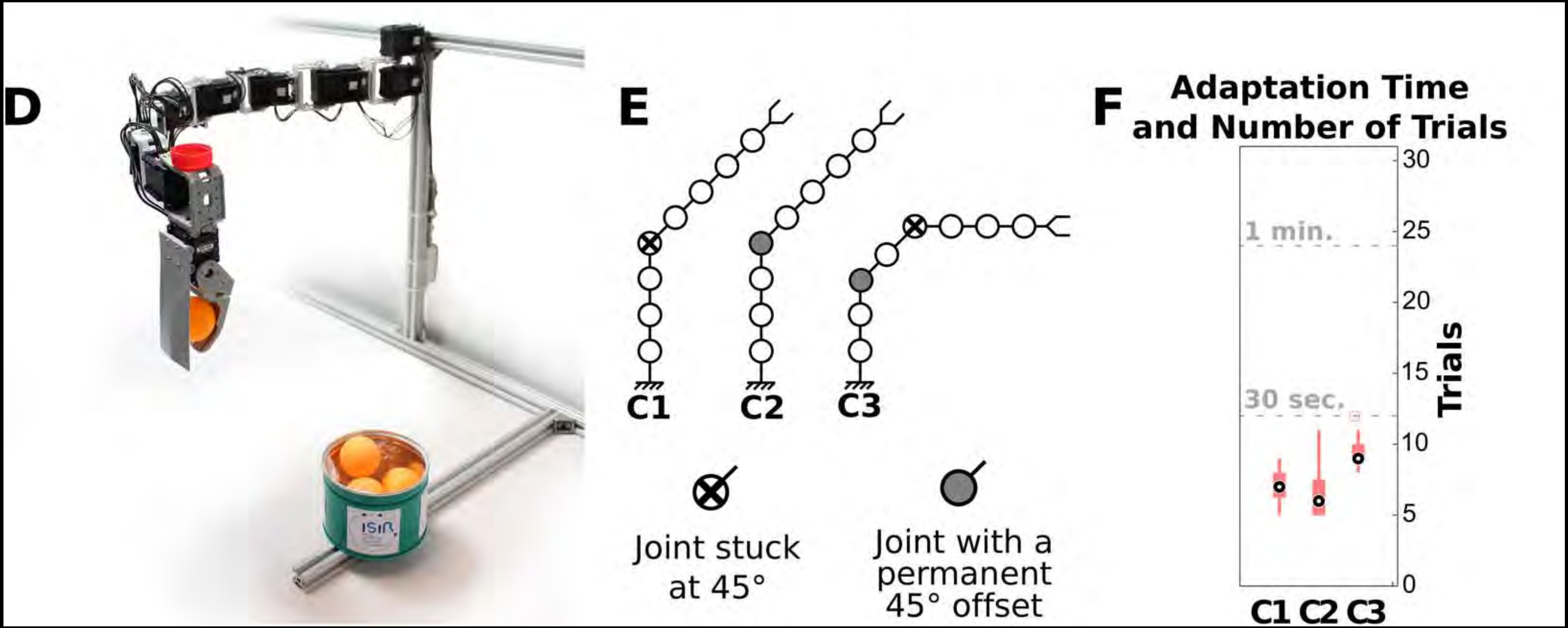
Different Damage Conditions & Behavioral Descriptions





Variant	Behavioral repertoire creation	Priors on performance	Search algorithm
Intelligent Trial and Error	MAP-Elites	yes	Bayesian Optimization
Variant 1	MAP-Elites	none	random search
Variant 2	MAP-Elites	none	Bayesian optimization
Variant 3	MAP-Elites	none	policy gradient
Variant 4	none	none	Bayesian optimization
Variant 5	none	none	policy gradient

Different Robot



Different Environments

Deep RL + Intelligent Trial & Error

- Policy gradients to optimize objective
- Store actions in each bin
- Population-based policy gradients

Map-based Multi-Policy Reinforcement Learning: Enhancing Adaptability of Robots by Deep Reinforcement Learning

Ayaka Kume, Eiichi Matsumoto, Kuniyuki Takahashi, Wilson Ko and Jethro Tan

Abstract—**In order for robots to perform mission-critical tasks, it is essential that they are able to quickly adapt to changes in their environment as well as to injuries and or other bodily changes. Deep reinforcement learning has been shown to be successful in training robot control policies for operation in complex environments. However, existing methods typically employ only a single policy. This can limit the adaptability since a large environmental modification might require a completely different behavior compared to the learning environment. To solve this problem, we propose Map-based Multi-Policy Reinforcement Learning (MMPRL), which aims to search and store multiple policies that encode different behavioral features while maximizing the expected reward in advance of the environment change. Thanks to these policies, which are stored into a multi-dimensional discrete map according to its behavioral feature, adaptation can be performed within reasonable time without retraining the robot. An appropriate pre-trained policy from the map can be recalled using Bayesian optimization. Our experiments show that MMPRL enables robots to quickly adapt to large changes without requiring any prior knowledge on the type of injuries that could occur.**

A highlight of the learned behaviors can be found here: `https://youtu.be/QwInbilXNOE`.

I. Introduction

Humans and animals are well-versed in quickly adapting to changes in not only their surrounding environments, but also to changes to their own body, through previous experiences and information from their senses. Some example scenarios where such adaptation to environment changes takes place are walking in a highly crowded scene with a lot of other people and objects, walking on uneven terrain, or walking against a strong wind. On the other hand, examples of bodily changes could be wounds, incapability to use certain body parts due to task constraints, or when lifting or holding something heavy. In a future where robots are omnipresent and used in mission critical tasks, robots are not only expected to adapt to unfamiliar scenarios and disturbances autonomously, but also to recover from adversaries in order to continue and complete their tasks successfully. Furthermore, taking a long time to recover or adapt may result in mission failure, while external help might not be available or even desirable, for example in search and rescue missions. Therefore, robots need to be able to adapt to changes in both the environment and their own body state, within a limited amount of time.

Recently, deep reinforcement learning (DRL) has been shown to be successful in complex environments with both

All authors are associated with Preferred Networks Inc., Tokyo, Japan (e-mail:{kume, matsumoto, takahashi, wko, jettan}@preferred.jp)

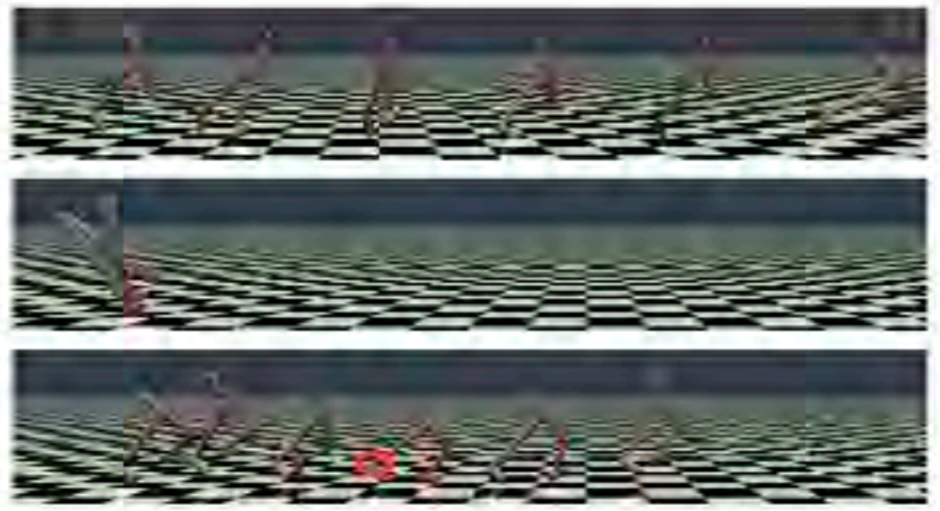


Fig. 1. Time lapse of the OpenAI Walker2D model walking for 360 time steps using a policy and succeeding while intact (top), failing due to a joint being limited (middle), and succeeding again post-adaptation despite the limited joint marked in red by selecting an appropriate policy using our proposed method (bottom).

high-dimensional action and state spaces [1], [2]. The success of these studies relies on a large number of samples in the orders of millions, so re-training the policy after the environment change is unrealistic. Some methods avoid re-training by increasing the robustness of an acquired policy and thus increasing adaptability. In robust adversarial RL, for example, an agent is trained to operate in the presence of a destabilizing adversary that applies disturbance forces to the system [3]. However, using only a single policy limits the adaptability of the robot to large modifications which requires completely different behaviors compared to its learning environment.

We propose Map-based Multi-Policy Reinforcement Learning (MMPRL), which trains many different policies by combining DRL and the idea of using a behavior performance map [4]. MMPRL aims to search and store multiple possible policies which have different behavioral features while maximizing the expected reward in advance in order to adapt to the unknown environment change. For example, there are various ways for multi-legged robots to move forward: walking, jumping, running, side-walking, etc. In this example, only the fastest policy would survive when using ordinary RL, whereas MMPRL saves all of them as long as they have different behavioral features. These policies are stored into a multi-dimensional discrete map according to its behavioral feature. As a result, adaptation can be done within reasonable time without re-training the robot, but just by searching an appropriate pre-trained policy from the map using an efficient method like Bayesian optimization, see Figure 1. We show that, using MMPRL, robots are able to quickly adapt to large changes with little knowledge about what kind of accidents will happen.

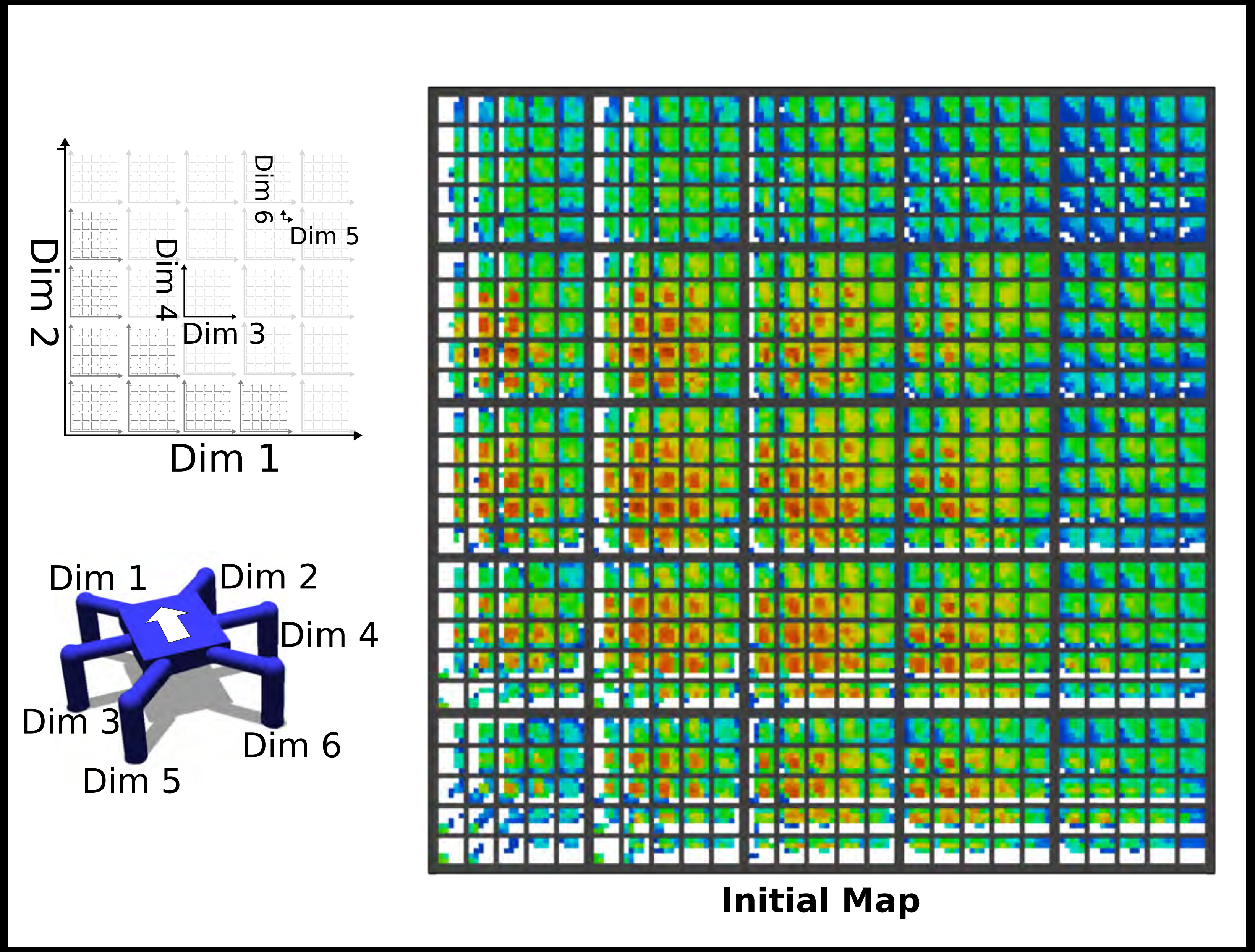
Conclusions: Intelligent Trial & Error

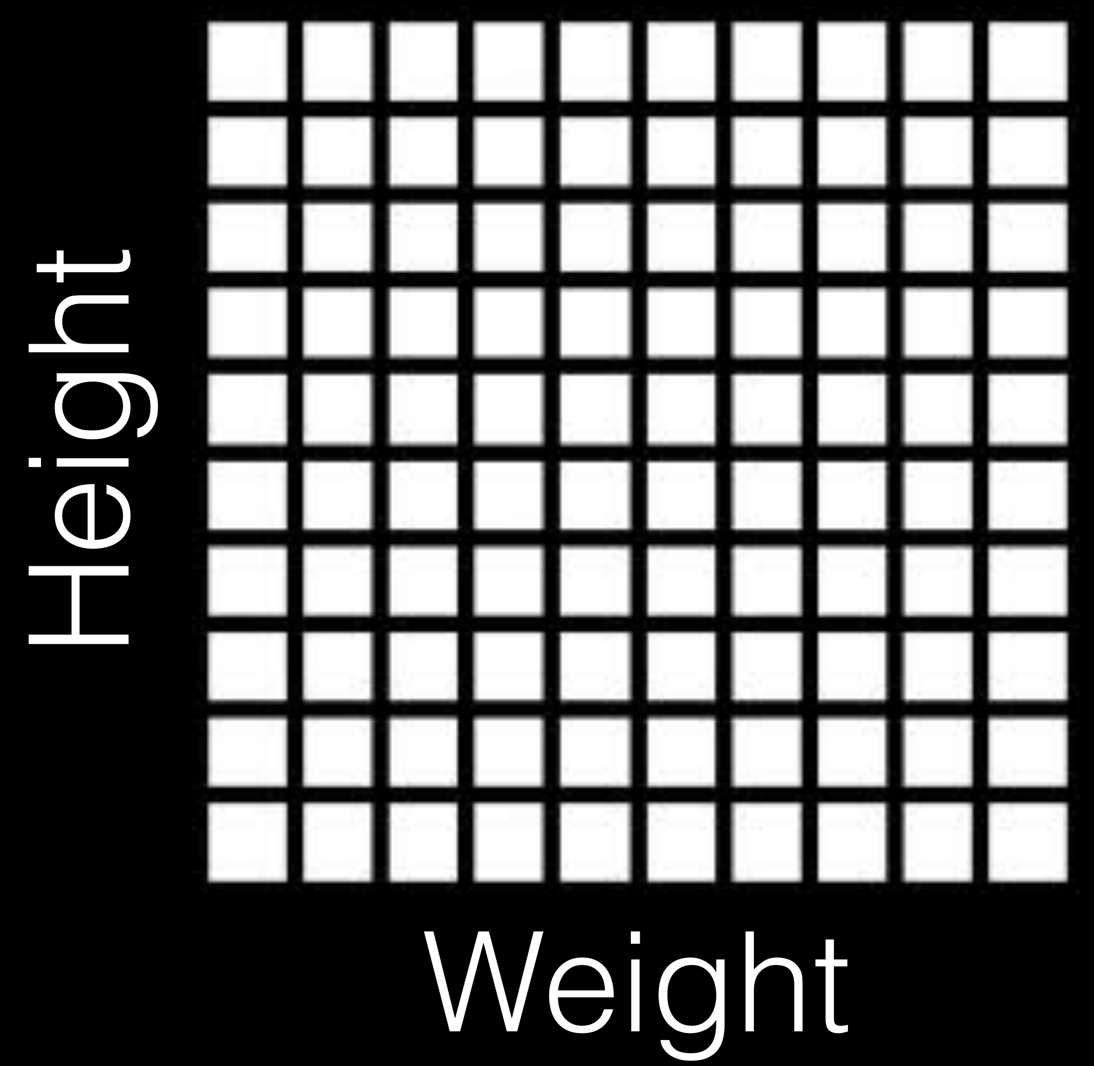
- State of the Art Robot Damage Recovery
  - adaptation, more broadly
- Adapts in < 2 minutes
- Combines
  - expensive creativity/power of MAP-Elites (in simulation)
  - with data efficiency of Bayesian optimization (in the real world)
- Shows a benefit of QD: learning diverse, high-performing sets of policies

intuitions about different ways to move **MAP-Elites** → few, intelligent tests **Bayesian Optimization** → pick one that works despite injury **found > X% of best**

Behavioral Characterization

- Hand-coded in most work

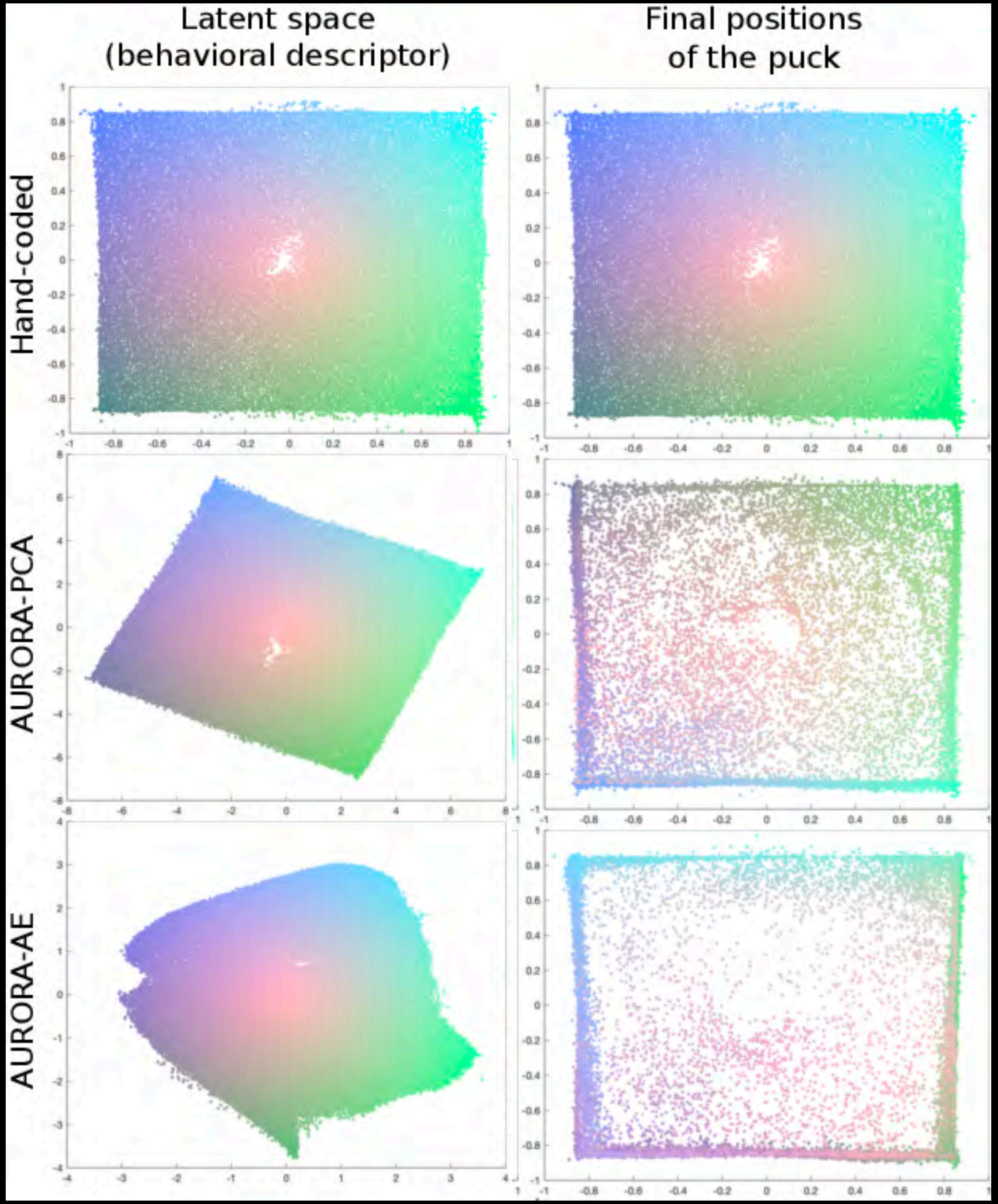




Learned Behavioral Characterizations

AURORA, Cully 2019

- Generate data randomly
- Loop
  - Apply dimensionality reduction
    - e.g. auto-encoder
  - Discretize latent code
  - Run MAP-Elites



Go-Explore

A new approach for hard-exploration problems

Adrien Ecoffet

Joost Huizinga

Joel Lehman

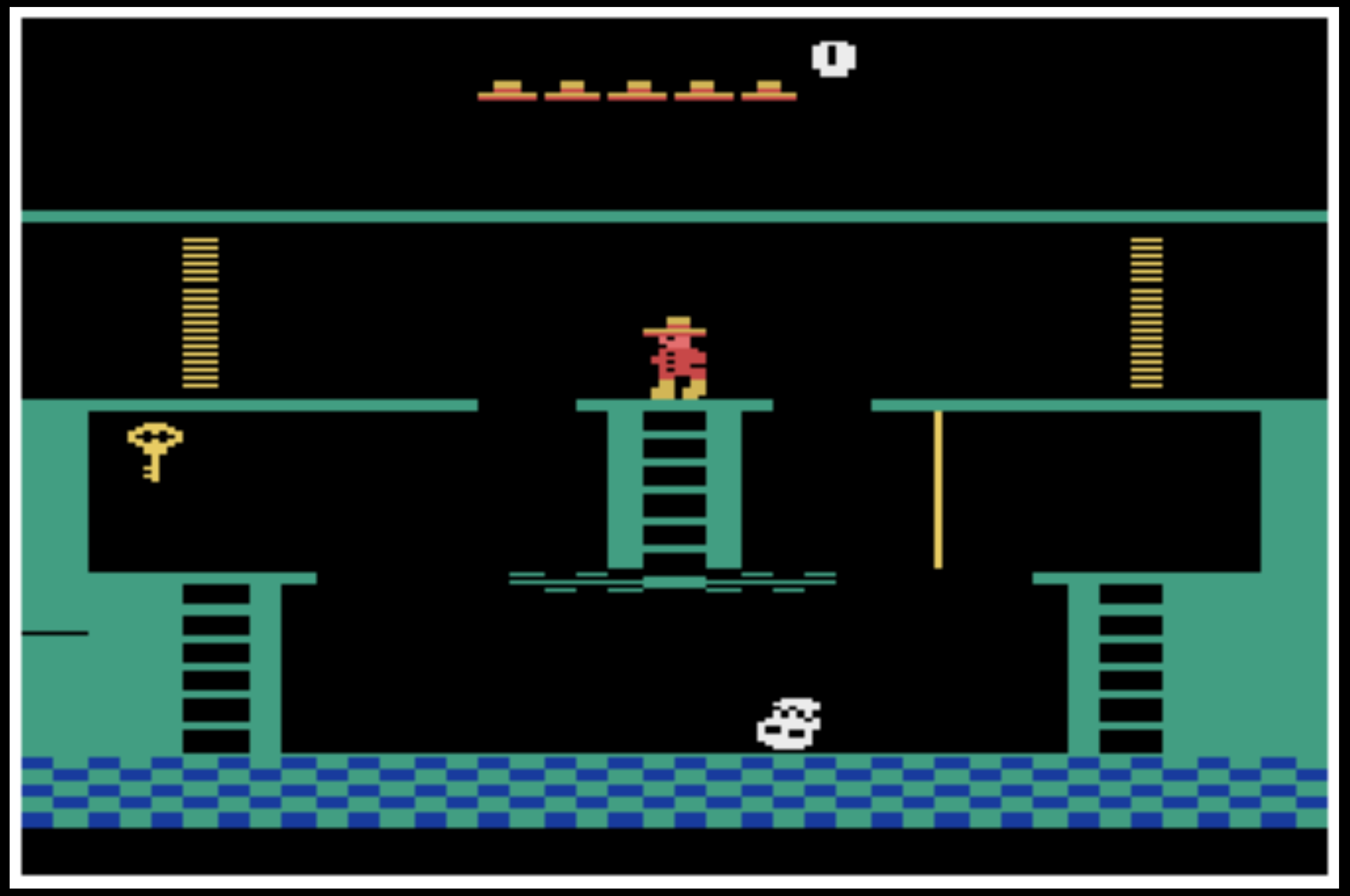
Ken Stanley*

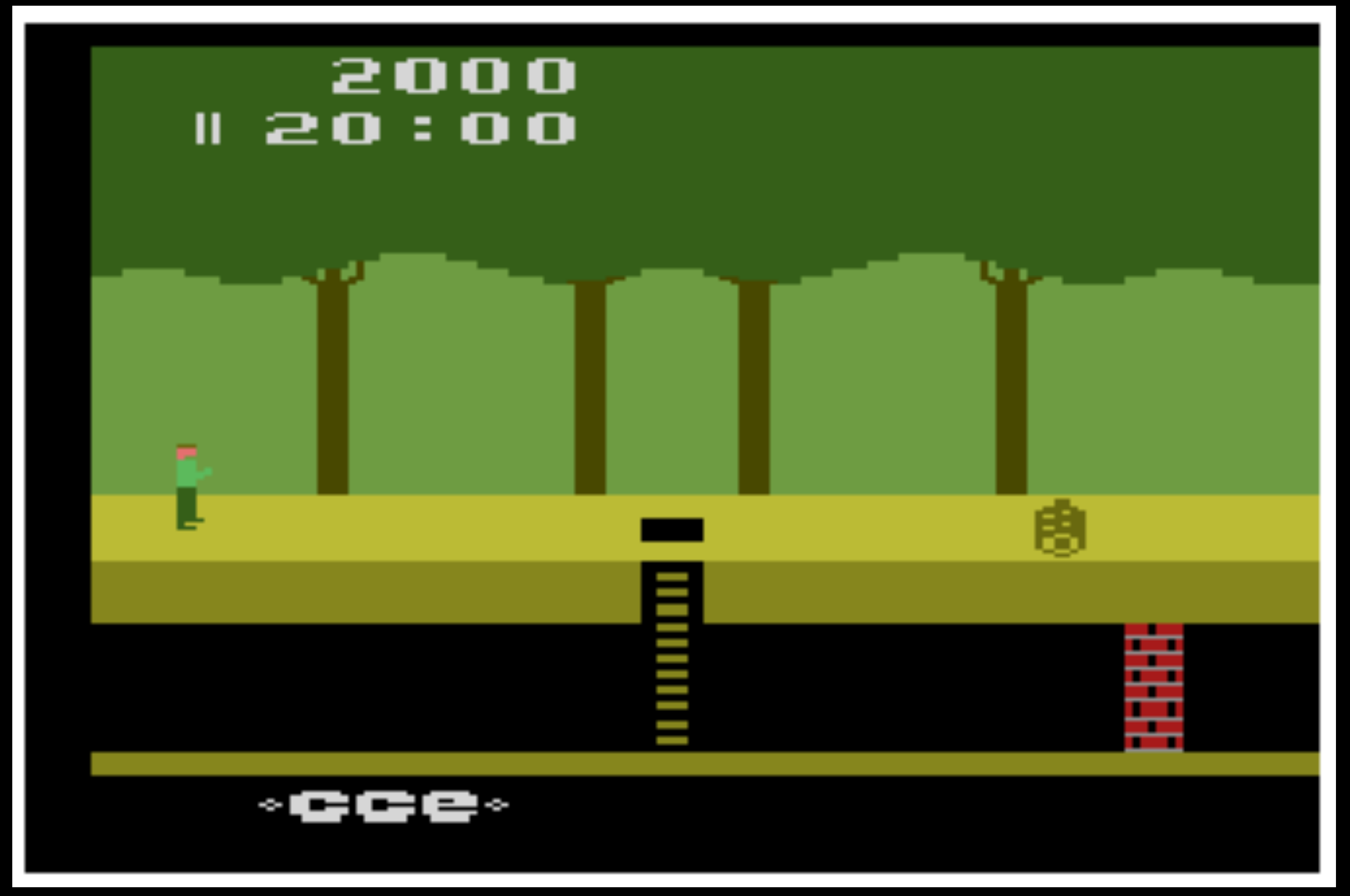
Jeff Clune*

UBER AI Labs

Grand Challenge in Deep RL
Effective Exploration

- Hard-exploration problems
  - Sparse-reward problems
    - rare feedback
    - Montezuma's Revenge
  - Deceptive problems
    - wrong feedback (wrt global optimum)





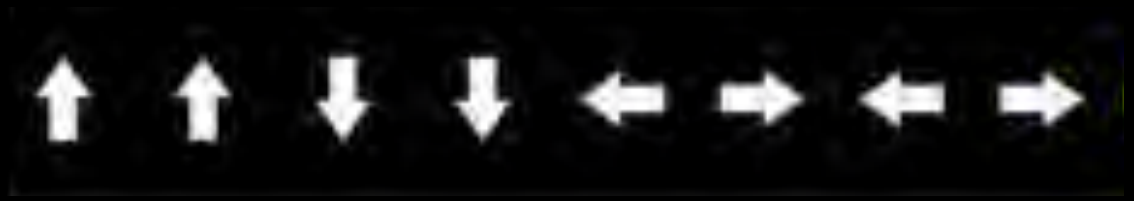
Go-Explore

Separates learning a solution into two phases

Phase 1: Explore Until Solved

Select state from archive → Go to state → Explore from state → Update archive

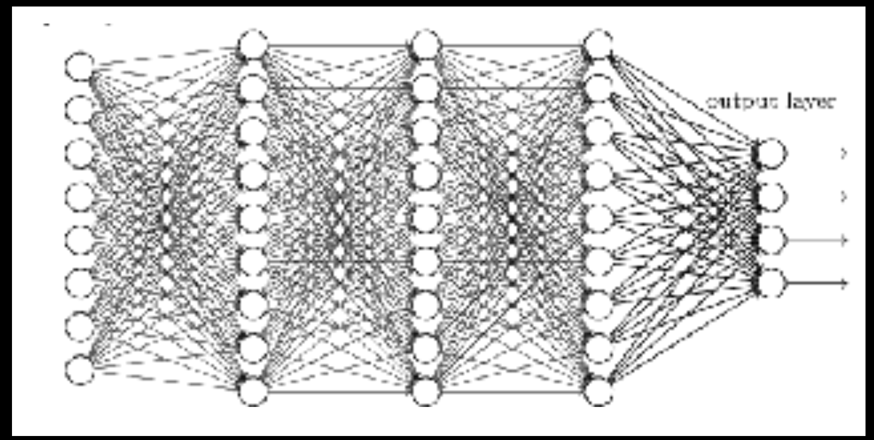
current work:
exploits deterministic training, no neural networks



Phase 2: Robustify
(if necessary)

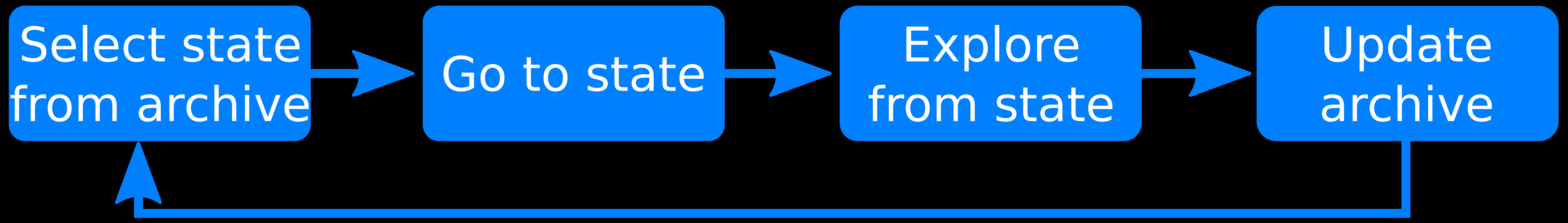
Run imitation learning on best trajectory

produces neural network
robust to stochasticity

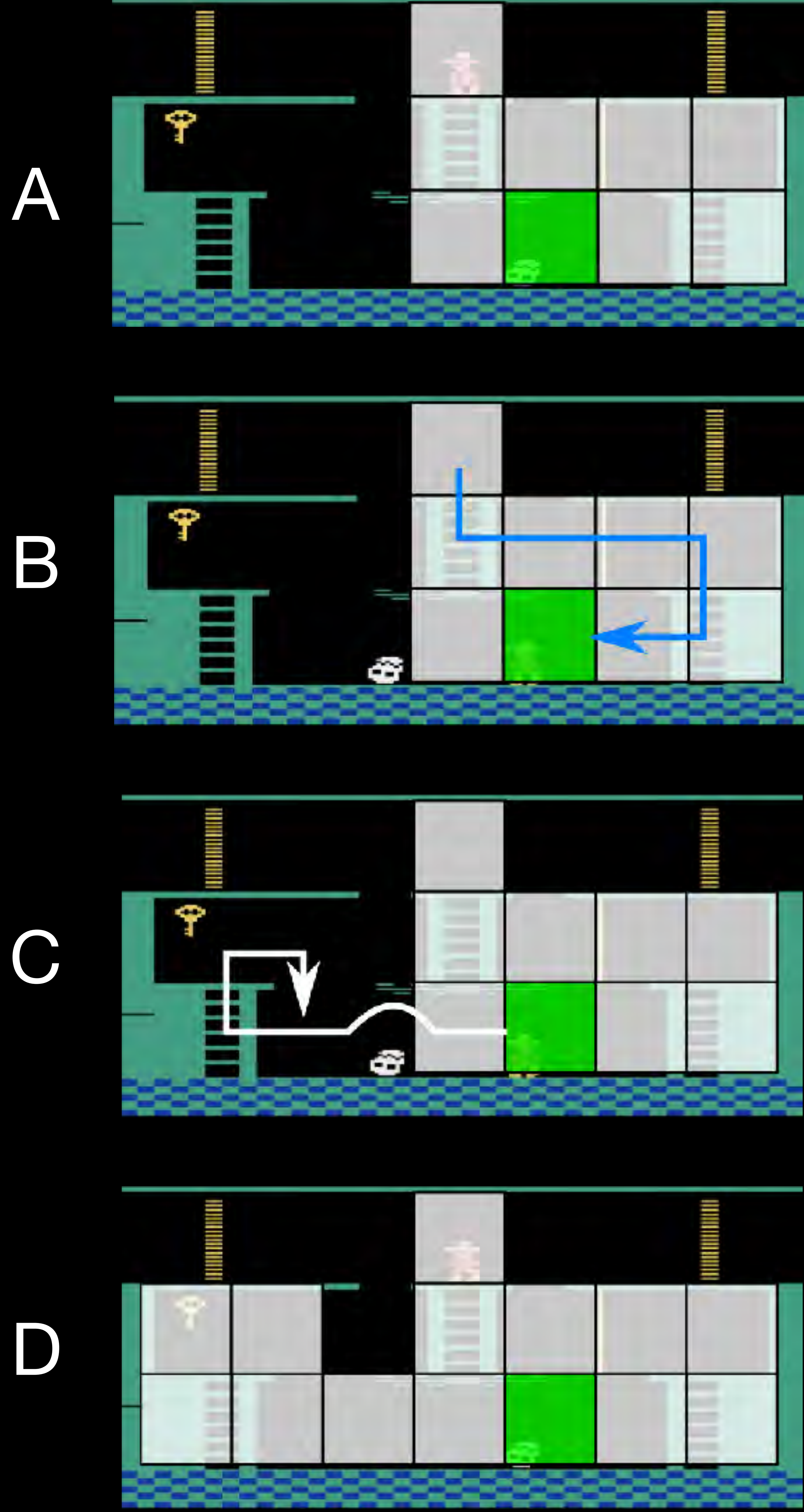


Go-Explore: Phase 1

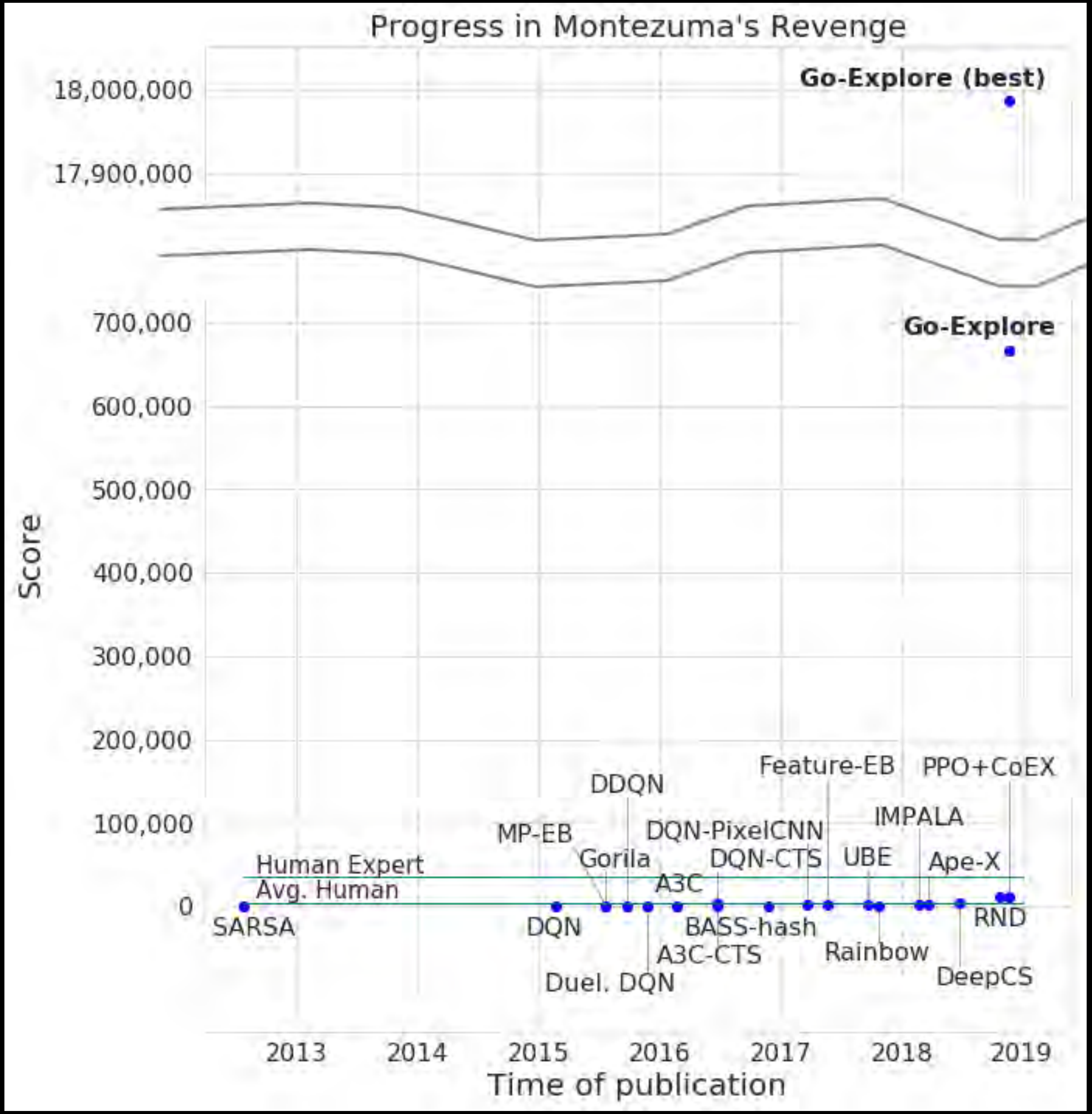
- Phase 1: explore until solved
  A. choose a state from archive
  B. Go back to it
  C. Explore from it
  D. add newly found states to archive
     - if better, replace old way of reaching state



An enhanced version of MAP-Elites



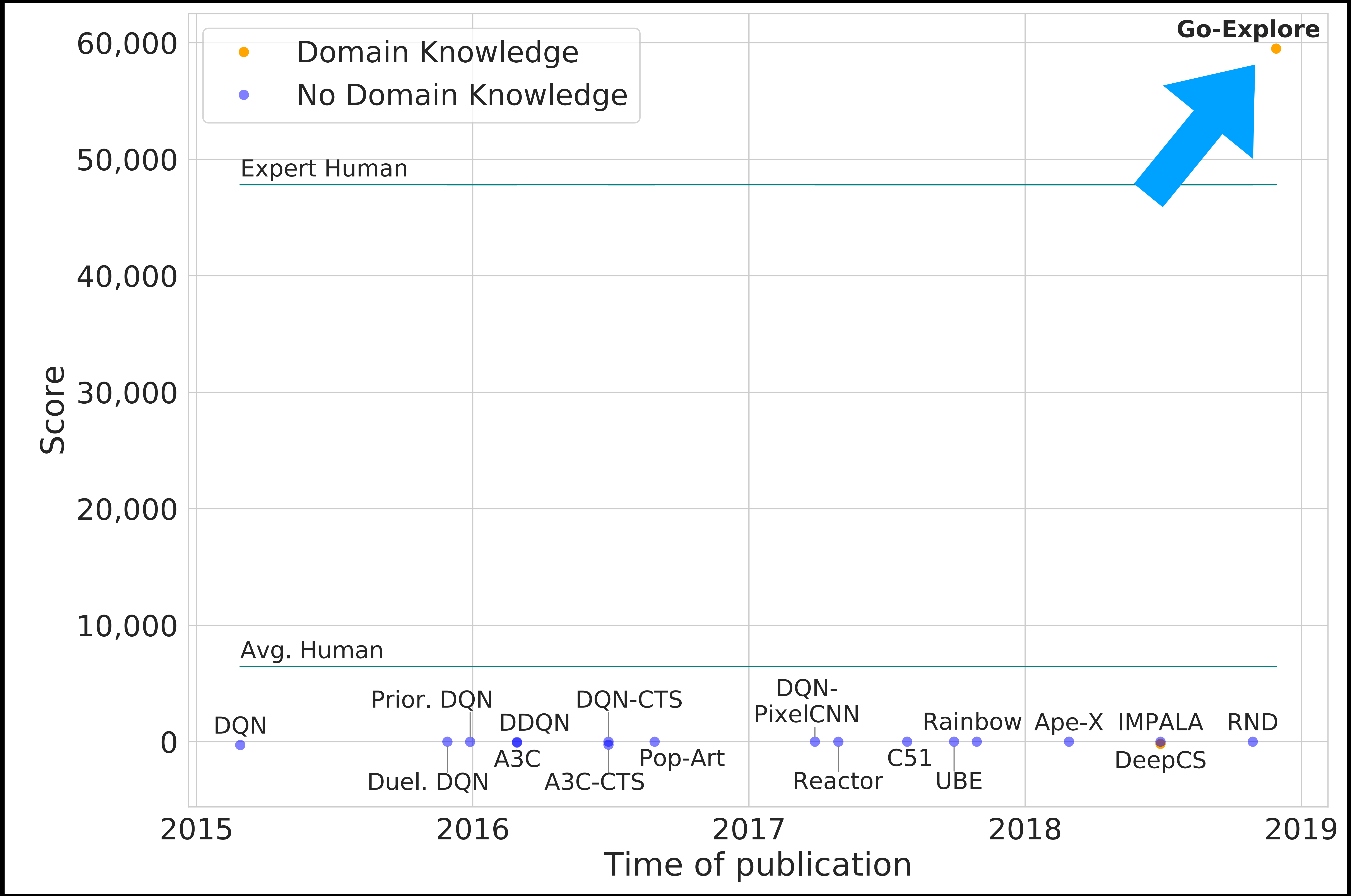
Montezuma's Revenge Results



- Average score: 660,000
- Best Go-Explore policy
  - scores ~18 million
  - solved 1,141 levels
- Beats human world record
  - 1,219,200

Note: exploits domain knowledge & deterministic training

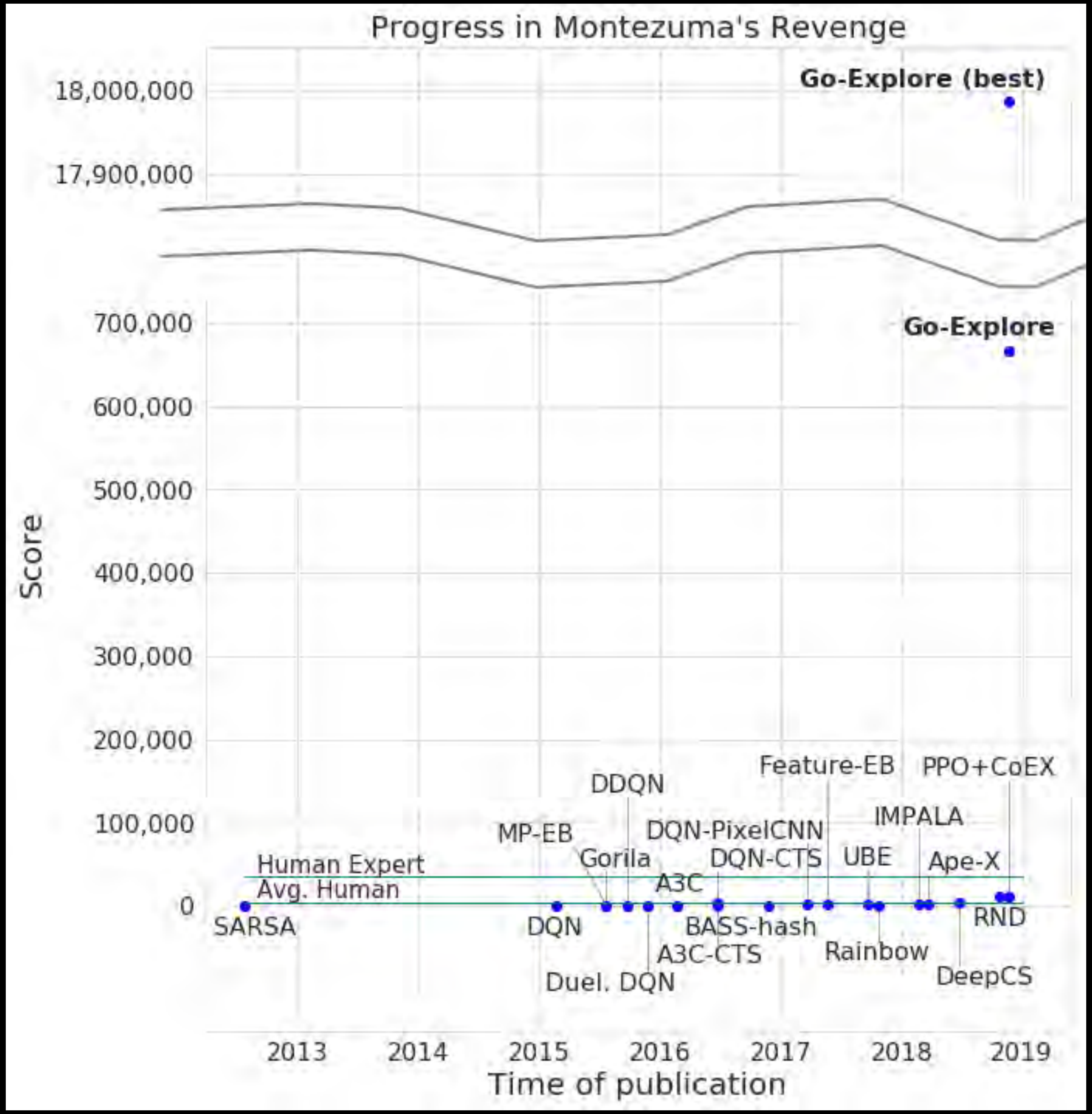
Pitfall Results



- no prior scores > 0
  - without:
    - fully deterministic test environment
    - or human demonstration
- average score: 59,000
- max: 107,000
- significantly advances state of the art

Go-Explore

- Shows value of QD ideas
  - collecting a diverse repertoire of high-quality entities
- Helped solve a previously unsolved problem

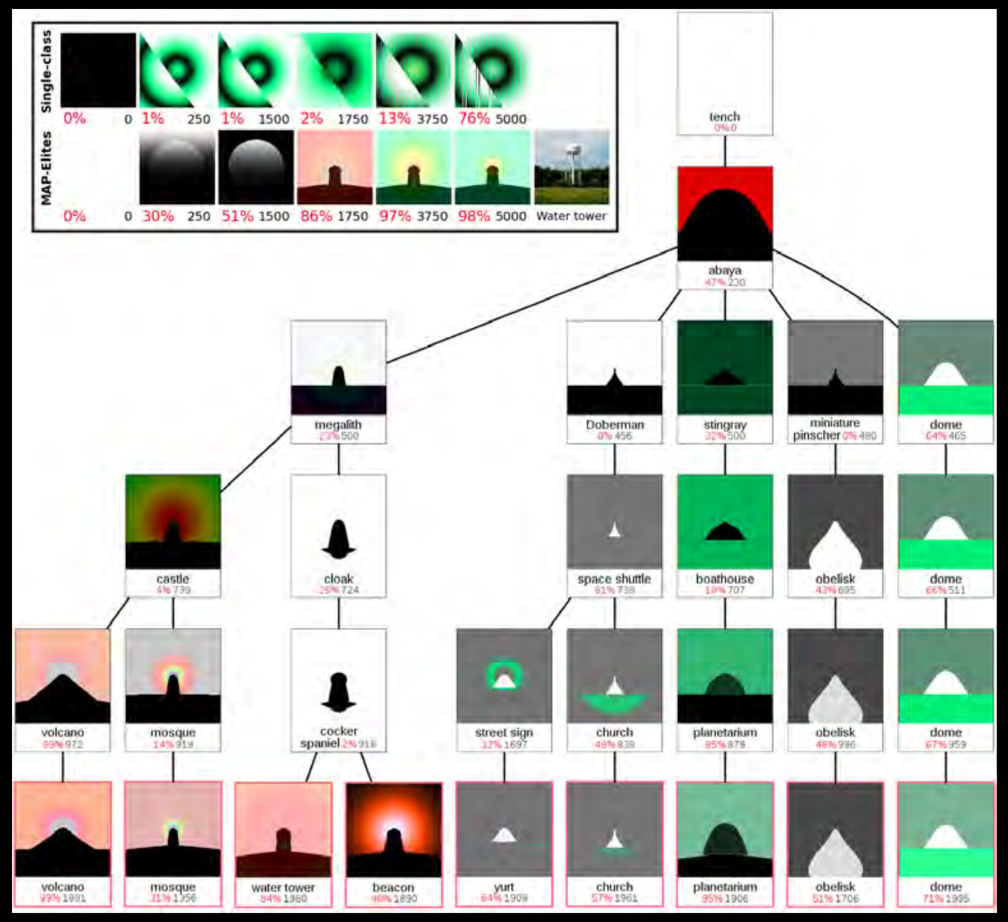


Future Work: Further Exploiting the QD Map

- Learn representations
- Learn world models
- Learn options (e.g. goal/task-conditioned policies)
- Learn agent models
- What else?

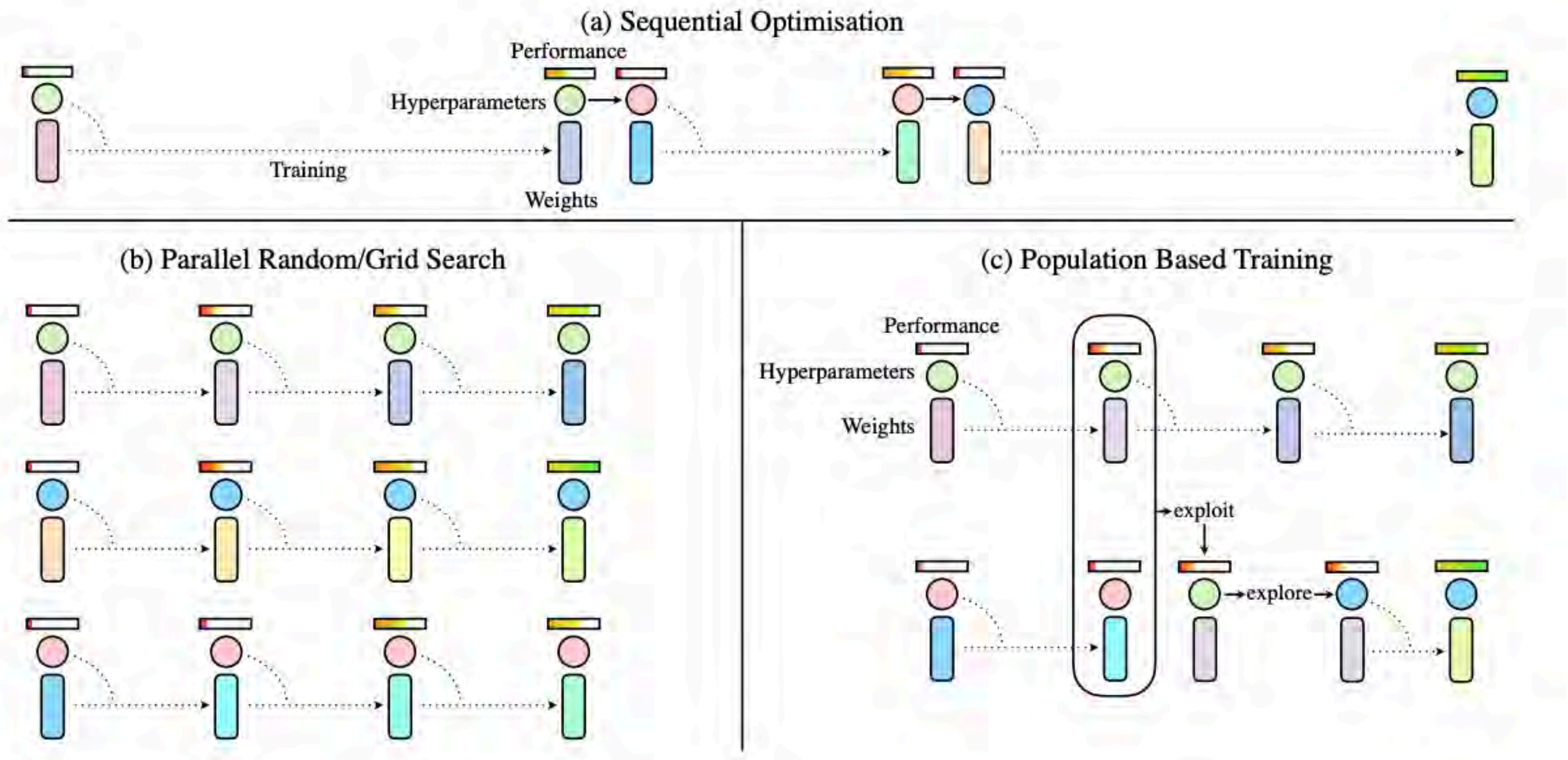
Conclusions: Quality Diversity Algorithms

- Generate a set of diverse, high-quality solutions
- Healthy internal dynamics
  - collect stepping stones
  - goal-switching
    - avoids local optima
    - harnesses serendipity
  - build on innovations via adaptive radiations
  - learn multiple, overlapping curricula
- Often is the best way even if you only want to solve one ambitious problem

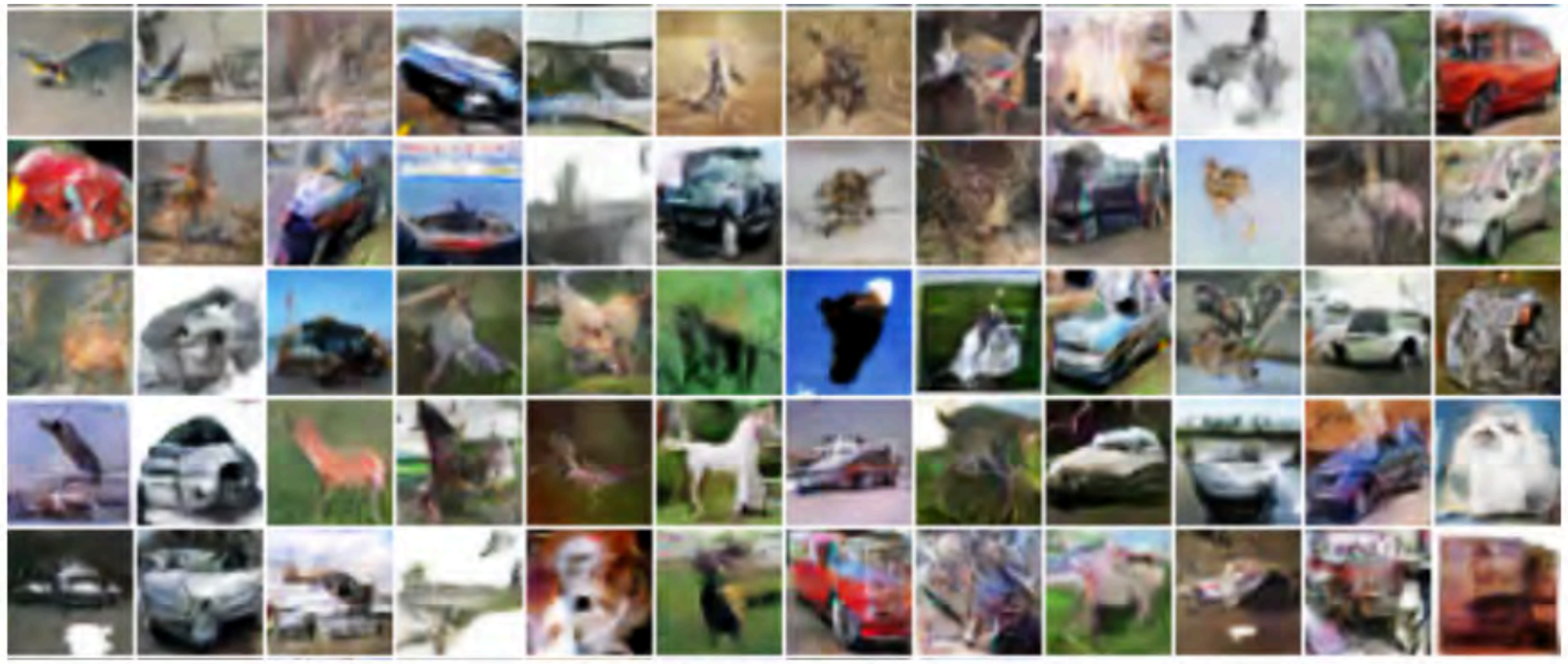


Related Work: Population Based Training + QD (inspired by Arulkumaran et al 2019)

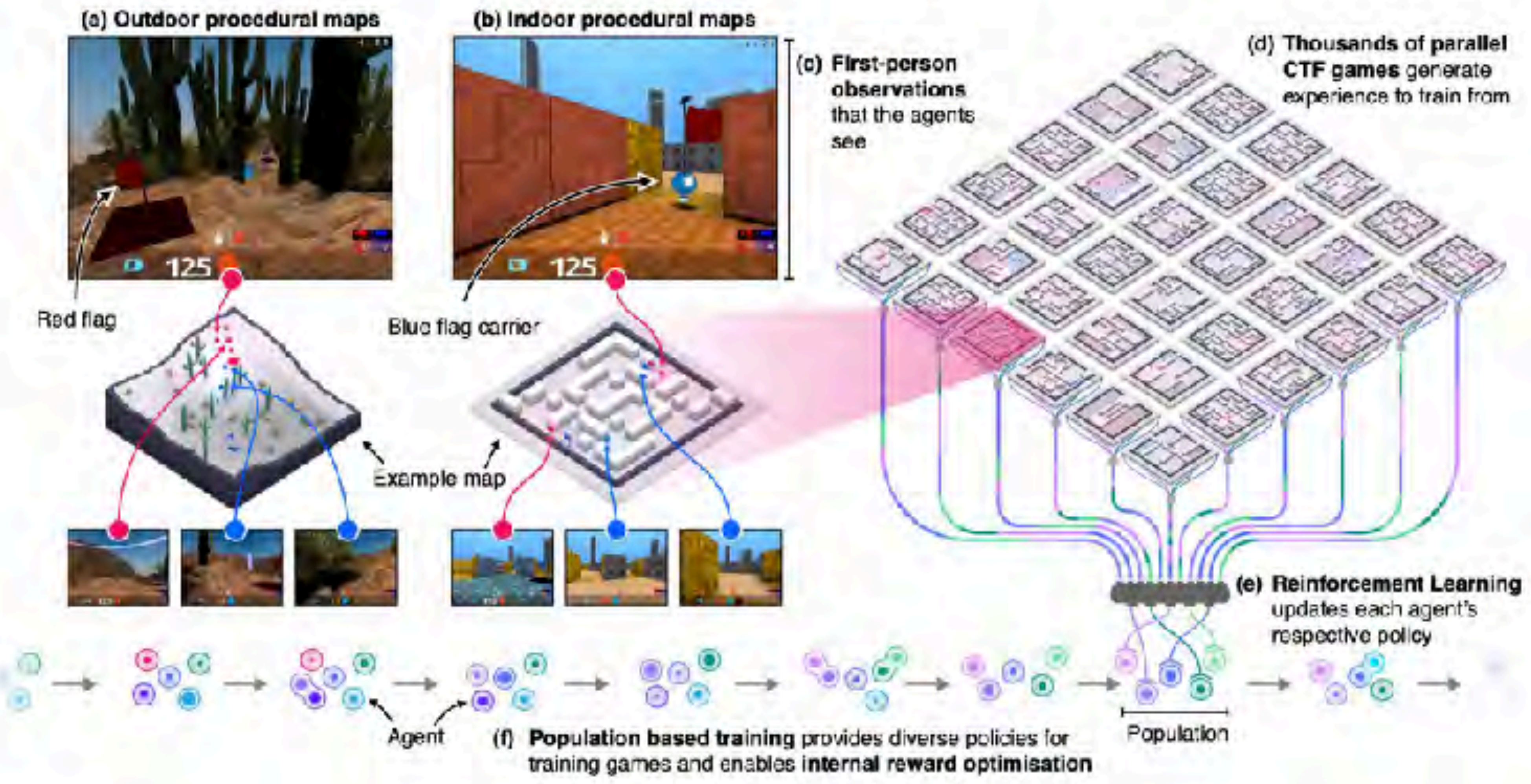
- Population-based training (Jaderberg et al. 2017)



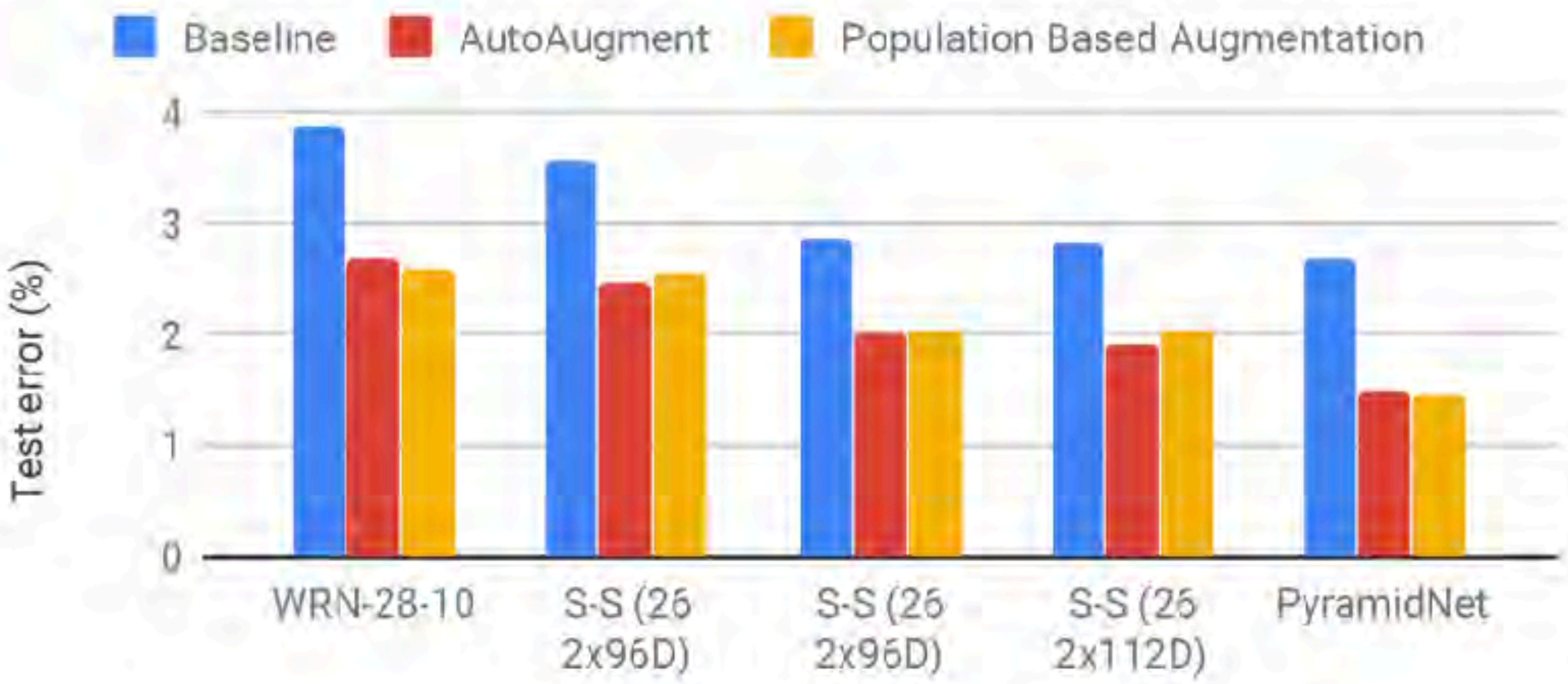
PBT Applications



PBT-GAN
(Jaderberg et. al 2017)



(Jaderberg et. al 2018)

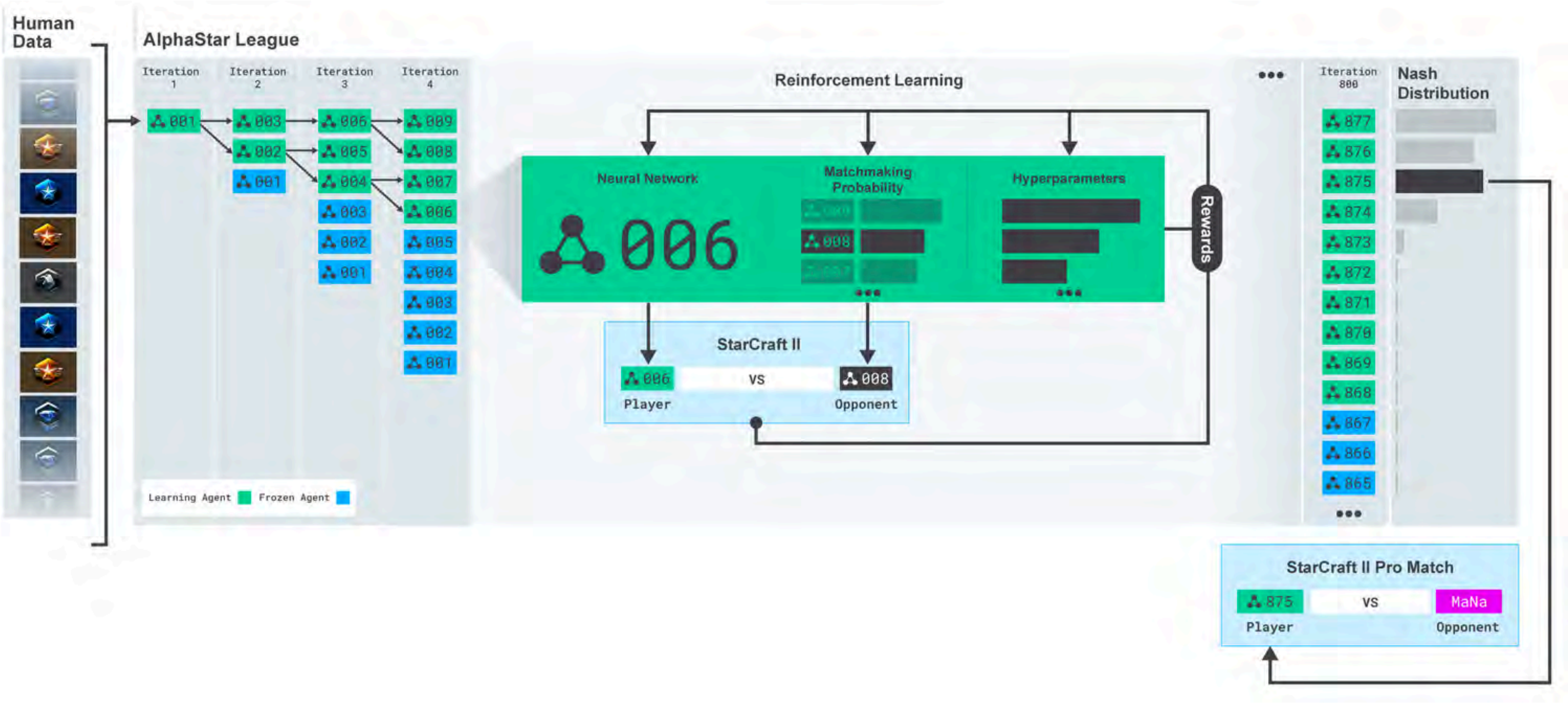


(Ho et. al 2019)

AlphaStar: Mastering the Real-Time Strategy Game StarCraft II

Games have been used for decades as an important way to test and evaluate the performance of artificial intelligence systems. As capabilities have increased, the research community has sought games with increasing complexity that capture different elements of intelligence required to solve scientific and real-world problems. In recent years, StarCraft, considered to be one of the most challenging Real-Time Strategy (RTS) games and one of the longest-played esports of all time, has emerged by consensus as a "grand challenge" for AI research.

Population Based Training + QD



Q&A

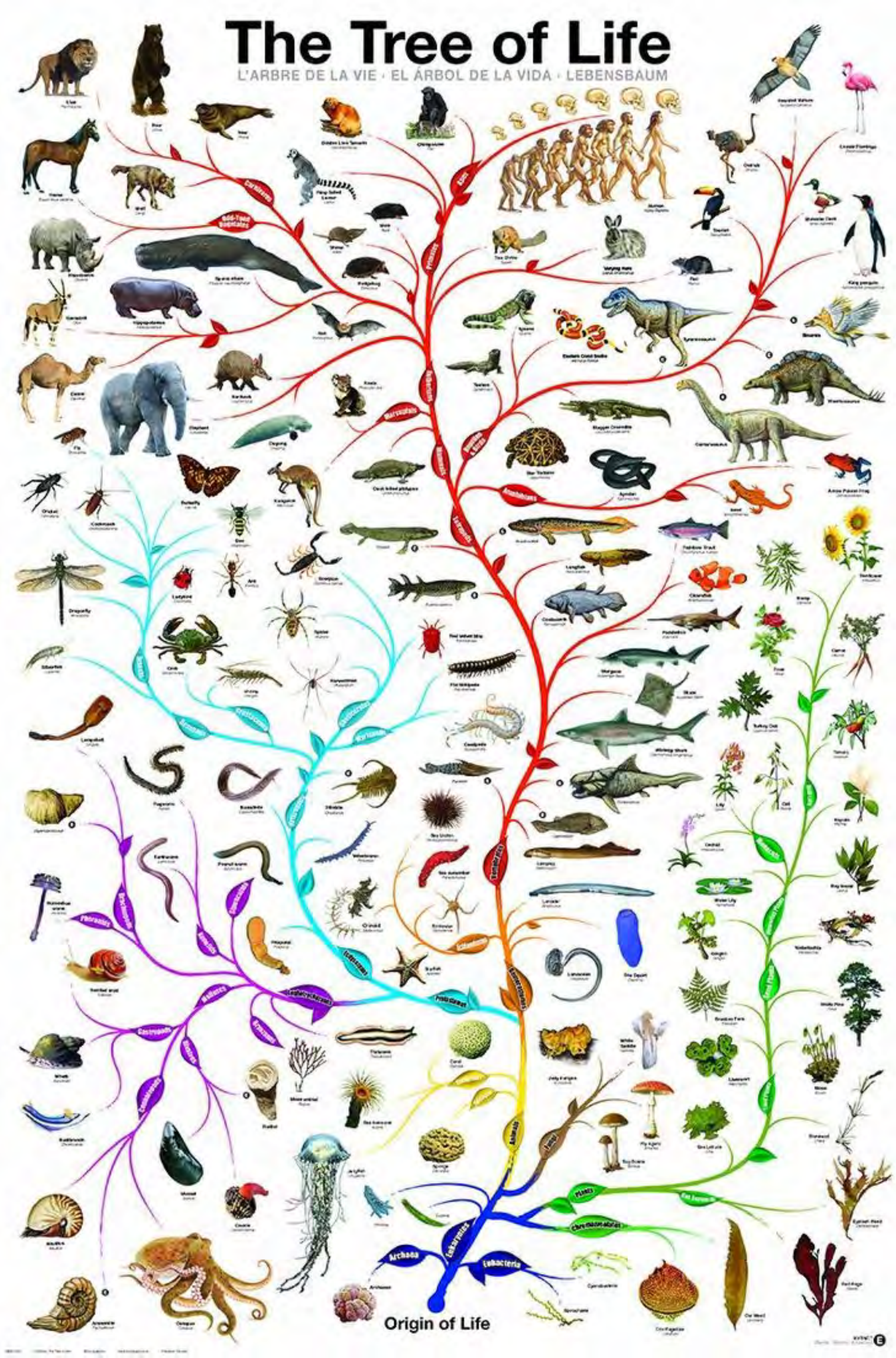
- 5 minutes

Beyond QD: The Grand Challenge of Open-Endedness

- Divergent search intentionally exposes the space of the possible
- But in any given domain, what is possible (at least of any interest), is finite
- Are there algorithms that not only find what is possible, but also *invent endless new possibilities*?
- QD seems close, but not quite there

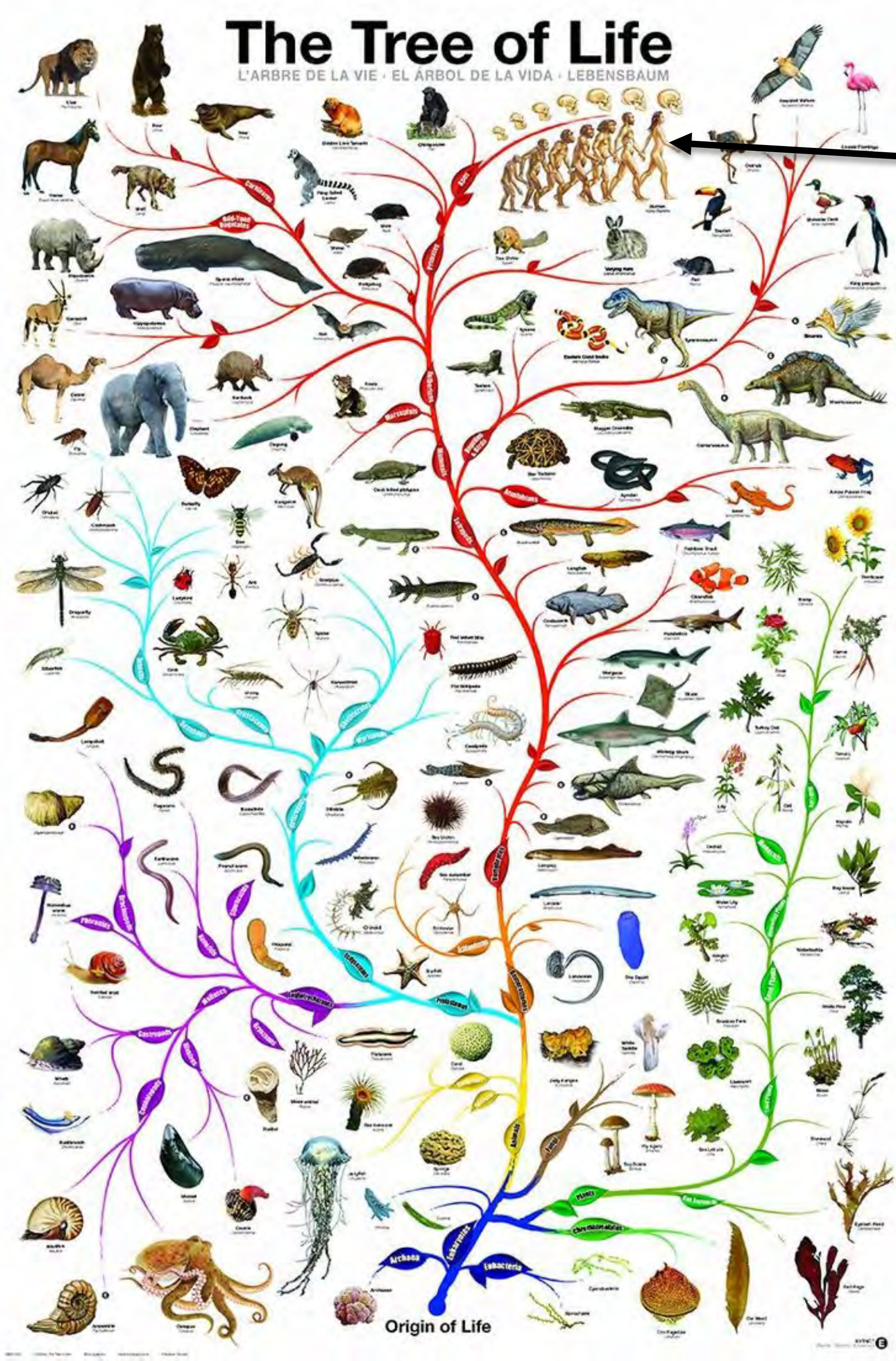
A Different Kind of Learning

- Not *how to learn something*
- But *how to learn everything*
- A human learning to play a video game is interesting
- But the history of human invention is *beyond interesting*
- Or: natural evolution – the ongoing creation of all the diversity of life on Earth



One run of **evolution**,
all life on Earth
(no human
intelligence!)

Thinglink.com



Human-level Intelligence, a tiny moment in an endless saga

One run of **evolution**, all life on Earth (no human intelligence!)

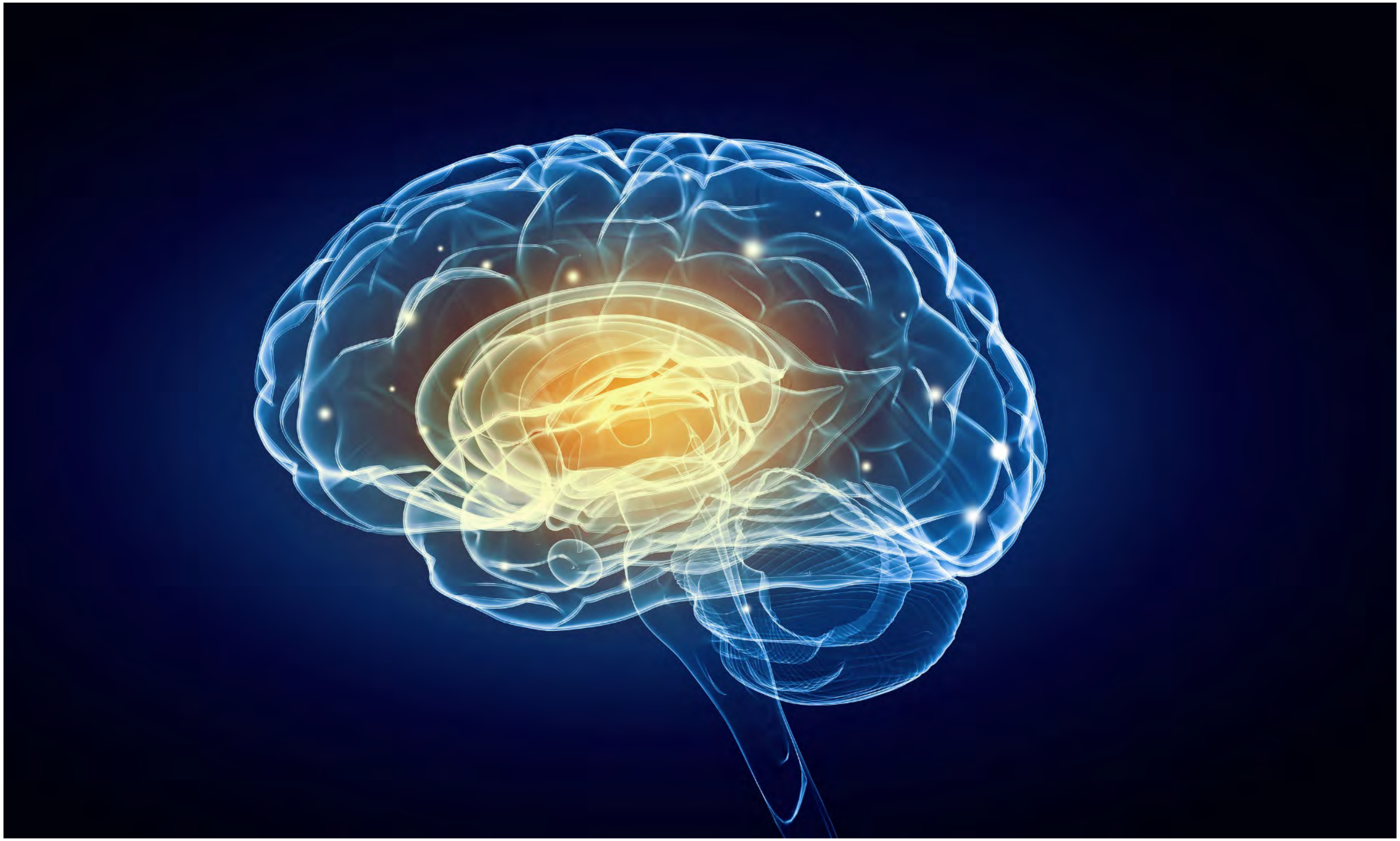
Thinglink.com

Endless Surprises!

(and it keeps on going)

One run of **evolution**, all life on Earth (no human intelligence!)

Thinglink.com



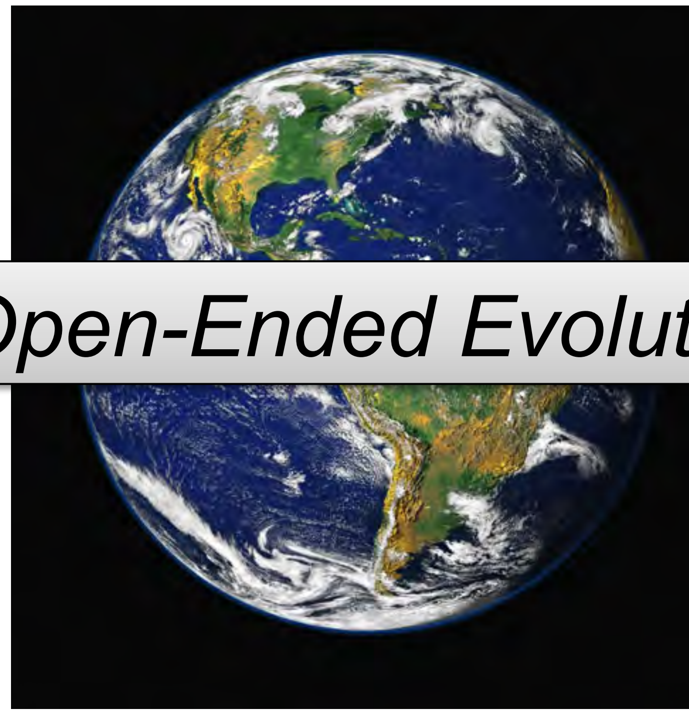
Not Like Even the Closest Ideas

- Not like QD
  - QD doesn't invent new problems
- Not like a GAN
  - A GAN exposed to billions of flatworms will never conceive a human
- Not like self-play or coevolution
  - AlphaGo will only improve at Go
  - There will never be a new game in town
- What kind of algorithm is OE?

The Never-Ending Algorithm

bittbox.com

The Never-Ending Algorithm



Open-Ended Evolution

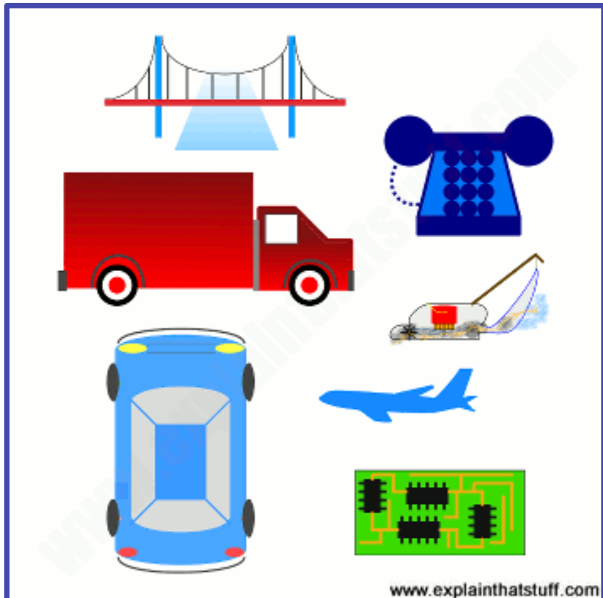
bittbox.com

The Never-Ending Algorithm

More Generally:
Open-Endedness

bittbox.com

The Never-Ending Algorithm



Open-Endedness:

The history of human innovation
…of art
…of science
…of architecture
etc…

Why don’t we create open-ended algorithms?

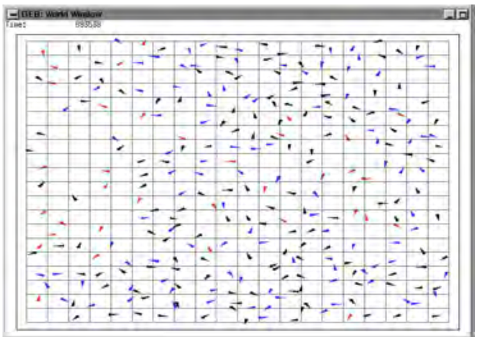
Why don't we create open-ended algorithms?

Why only solve problems?

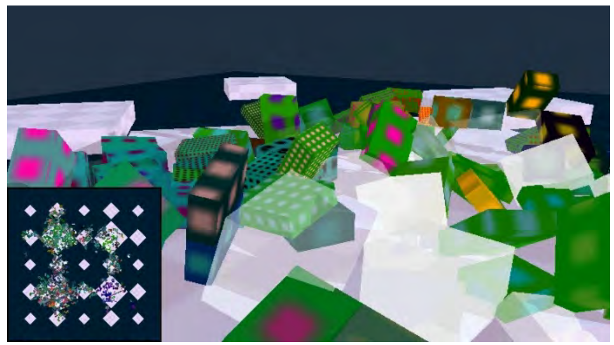
Exception: The OEE Community

- Open-ended evolution (OEE) is a traditional topic of artificial life
- OEE is the *power of creation*
  - Potentially transformative
  - Boundless creativity on demand
  - Discoveries beyond the scope of optimization
- A grand challenge on the scale of AI; maybe the path to AI itself
  - Why so little attention?

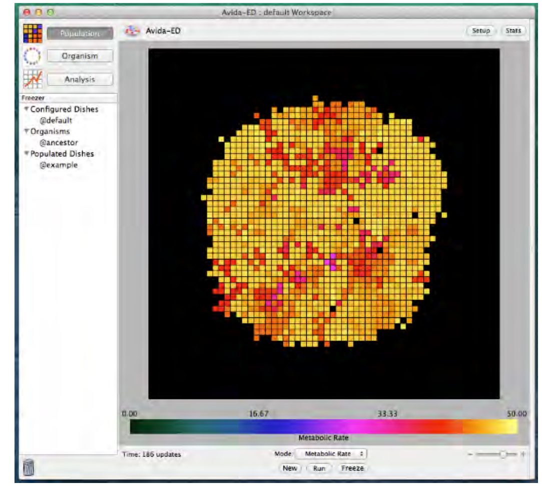
Much of the Seminal Work in Open-Endedness Was in "Alife Worlds"



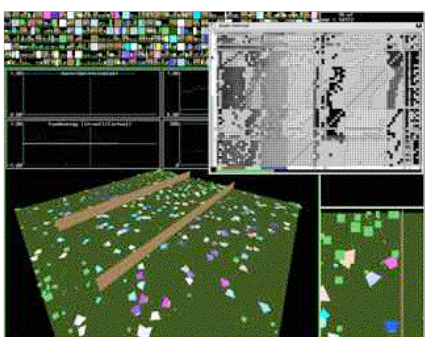
Geb
(Alastair Channon
2001, 2003)



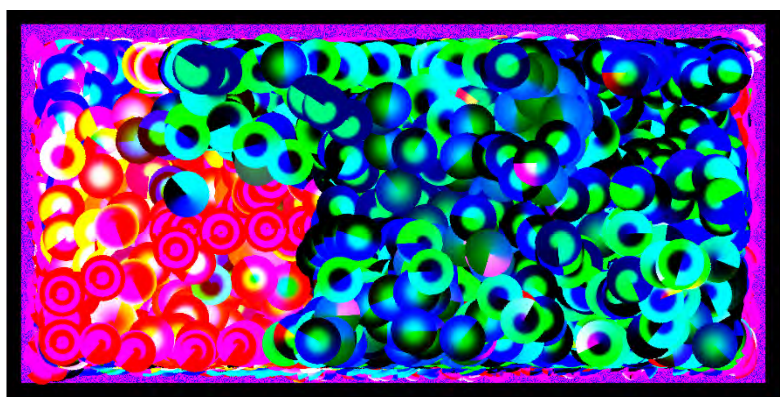
Division Blocks
(Lee Spector, Jon Klein,
Mark Feinstein 2007)



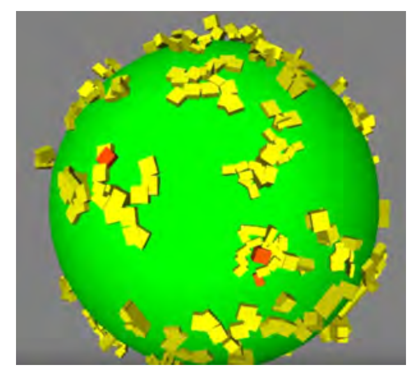
Avida (Charles Ofria, Chris Adami,
Titus Brown, et al. 1994-)



Polyworld (Larry Yaeger 1994-)



Chromaria (Lisa Soros & Ken Stanley 2014-)



Evosphere (Thomas Miconi 2008)

But It Doesn’t Have to Be a “World”

- A “world” is just a conduit to understanding
- It doesn’t even have to be a metaphor for organisms on Earth
  - Deep learning can play a role
- We are seeking the fundamental conditions for divergent, creative processes that never end
- They could be applied to *anything*

The Promise of Open-Endedness

- Design of buildings, vehicles, furniture, clothing, equipment, etc.
- Repertoires of controllers for vehicles, robots, UAVs, spaceships, etc.
- Endless generators of art and music
- Open-ended video game worlds with the granularity and originality of ecologies on Earth
- Renewed understanding and acceleration of the process of human invention
- Human-coupled open-ended systems
- Intelligence itself?

Even QD Algorithms Won’t Invent Forever

- Important step but…
- What happens when the space of the possible is filled?
- What causes *new* possibilities to arise?
  - And forever?
- Answer: The system needs to generate new opportunities *and* search through them at the same time
  - The key to Earth’s open-ended creativity

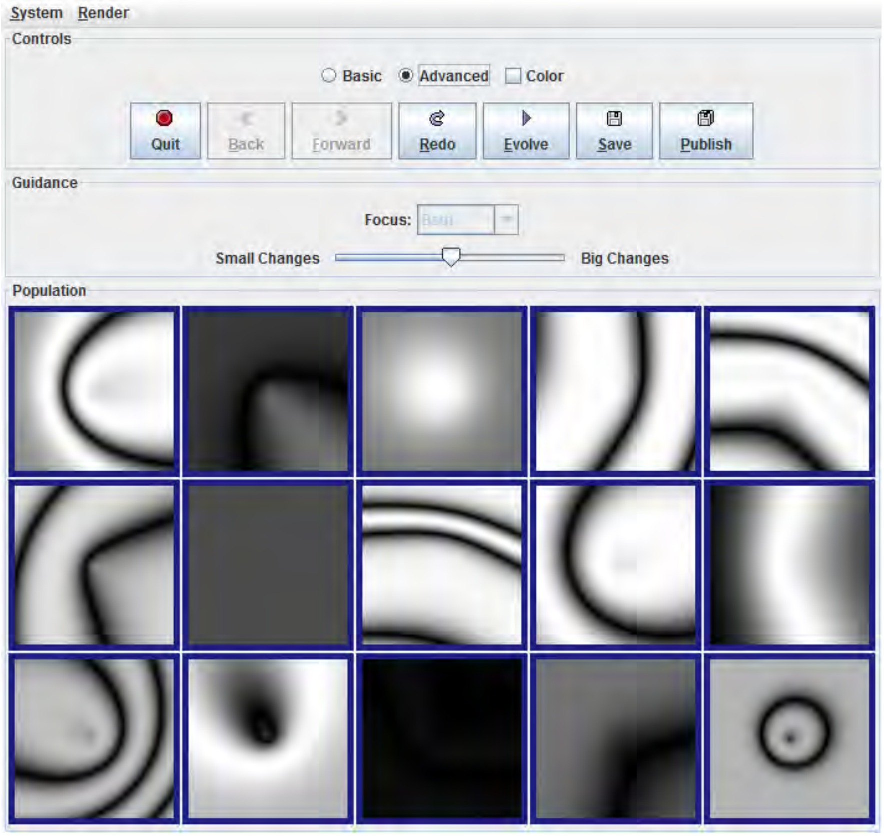
So How Will We Achieve Open-Endedness?

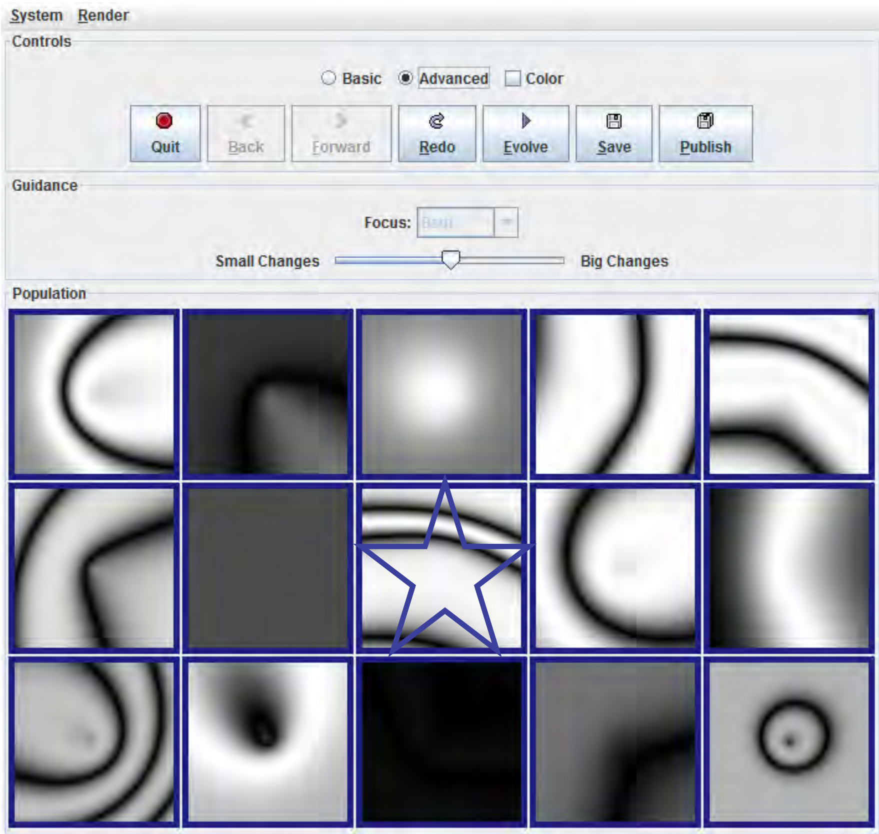
- Any great puzzle leads to surprises
  - Expect counter-intuitive insights

Some Interesting Clues in Artificial Systems

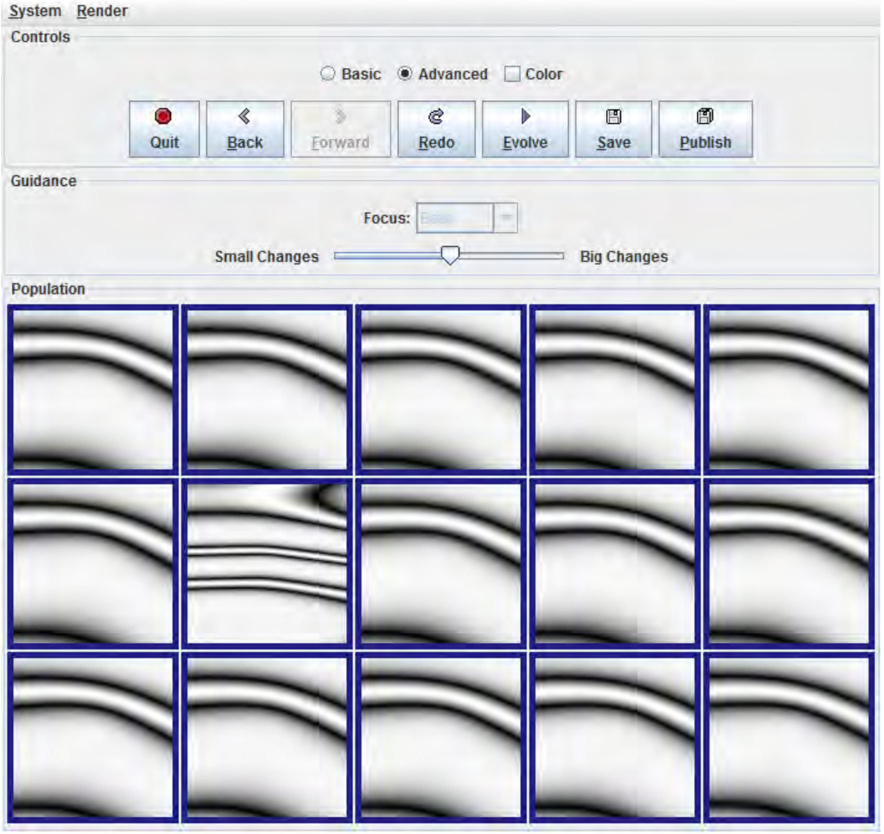
- The Picbreeder experiment
  - Showed actual signs of open-endedness
  - *But with humans in the loop*, breeding pictures
- Main idea: Anyone can follow up from anyone else's discoveries; no unified goal for the system

Observing Picbreeder.org

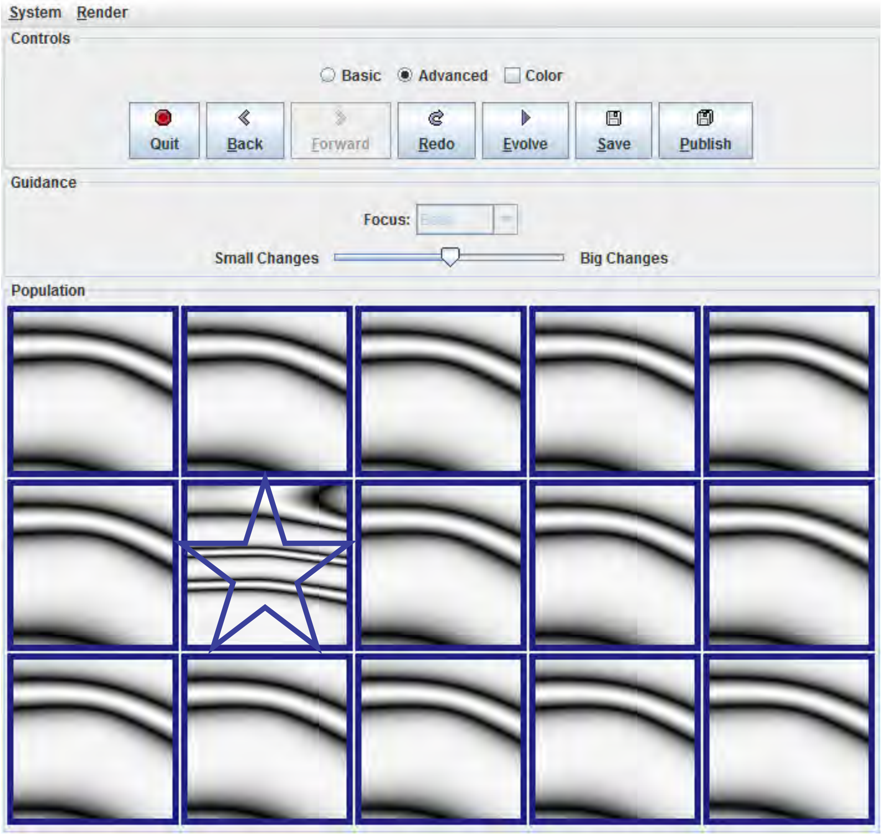



System
Render
Controls
Basic
Advanced
Color
Quit
Back
Forward
Redo
Evolve
Save
Publish
Guidance
Focus:
Small Changes
Big Changes
Population

Parent

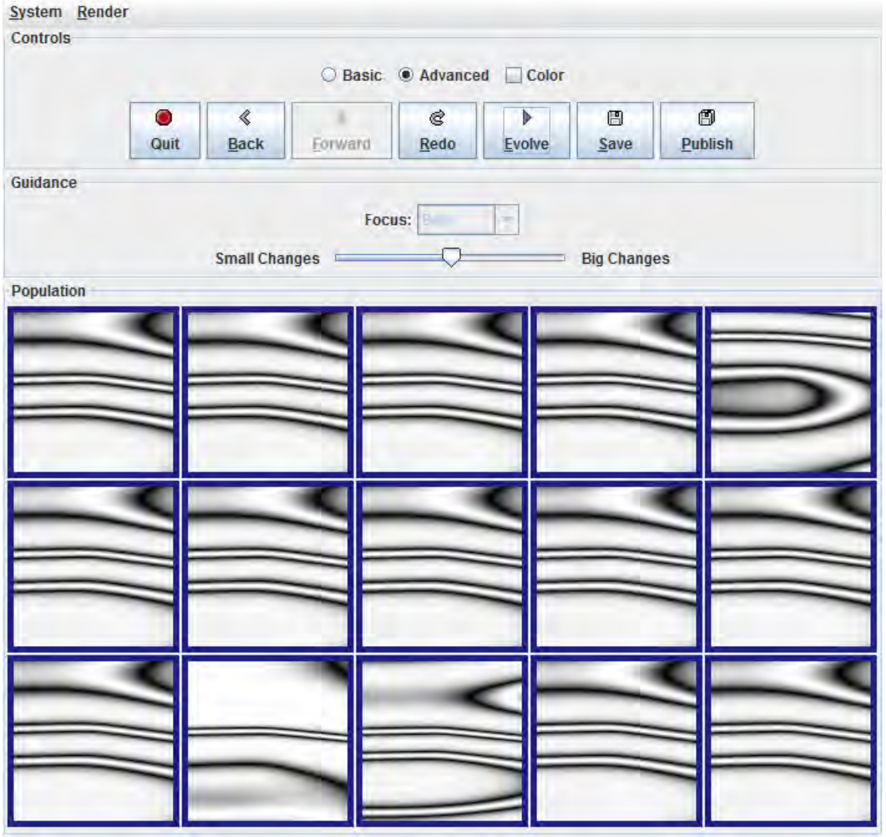

System
Render
Controls
Basic
Advanced
Color
Quit
Back
Forward
Redo
Evolve
Save
Publish
Guidance
Focus:
Small Changes
Big Changes
Population

Parent



System
Render
Controls
Basic
Advanced
Color
Quit
Back
Forward
Redo
Evolve
Save
Publish
Guidance
Focus:
Small Changes
Big Changes
Population

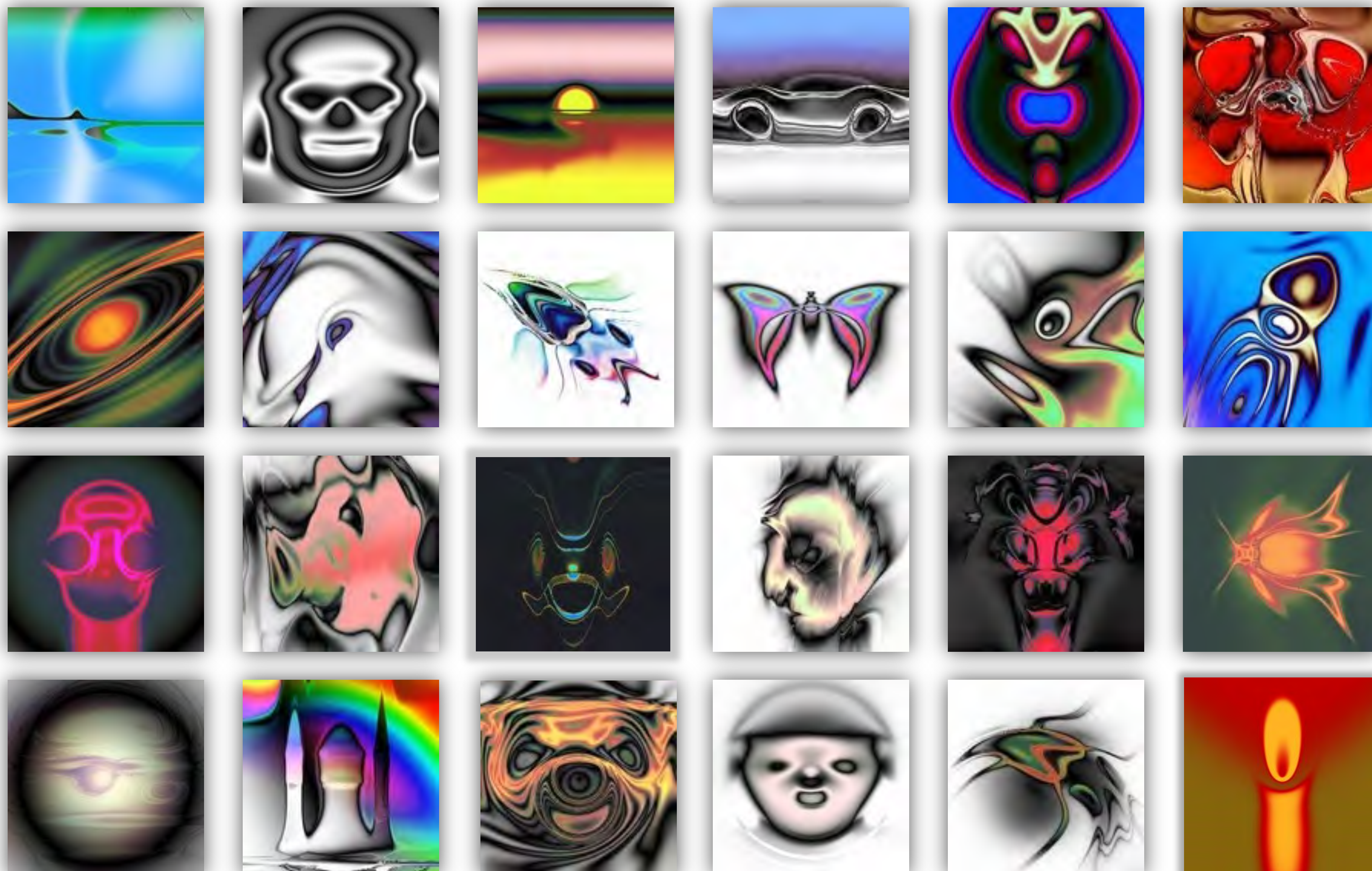
Parent



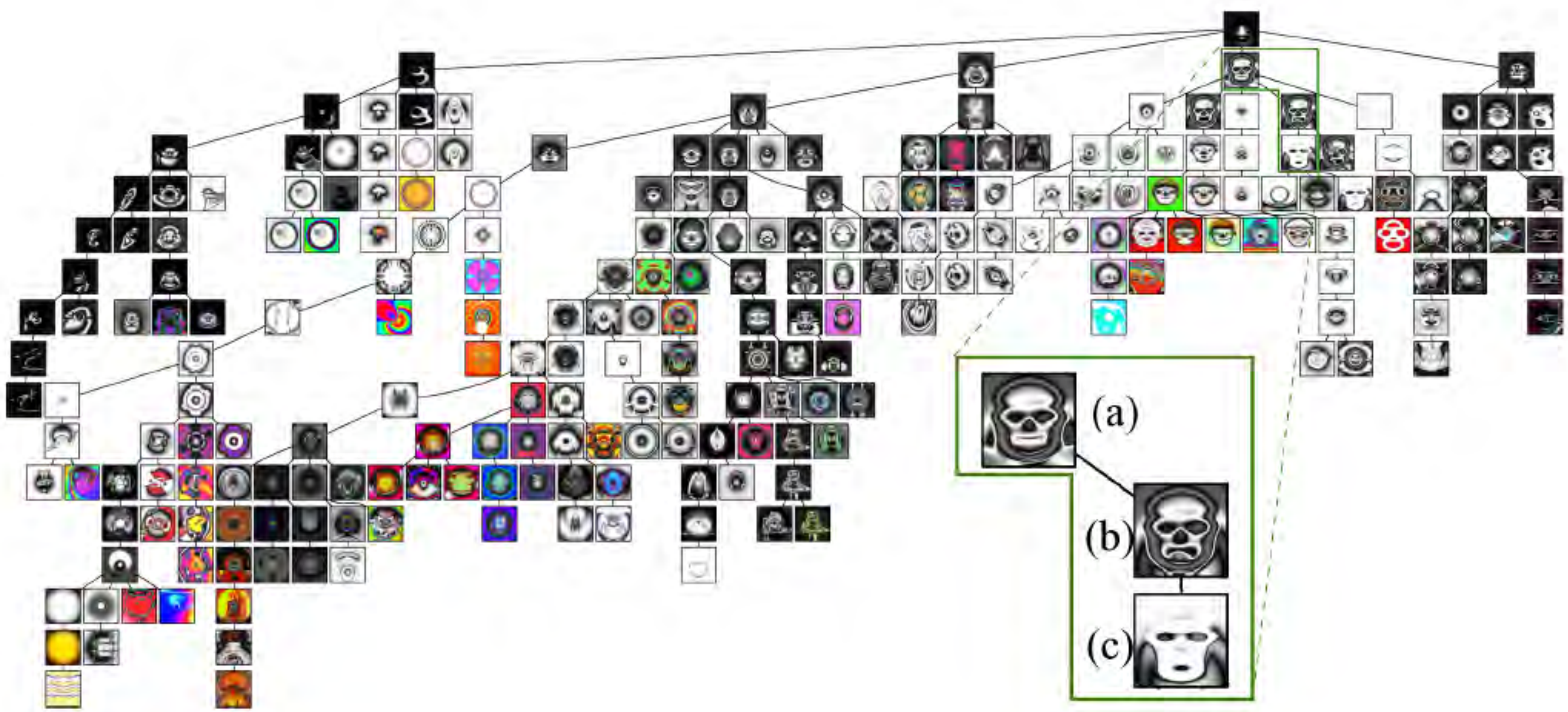
And
so on…

Discoveries by Picbreeder Users

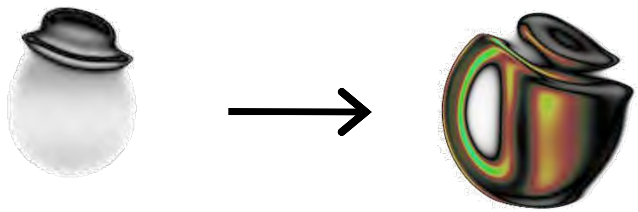
(All are 100% bred: no retouching)



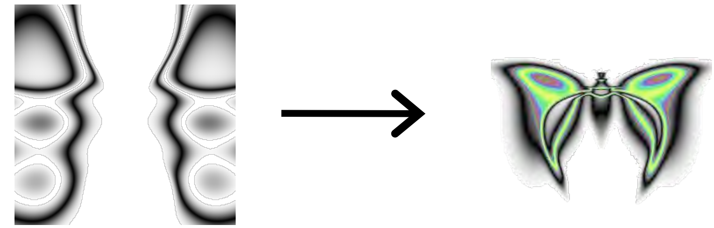
Actually Looks Open-Ended! (Phylogenies emerging)



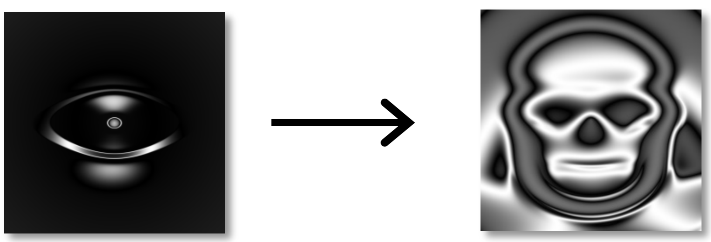
What We Discovered: People Only Find When They Are *Not* Seeking



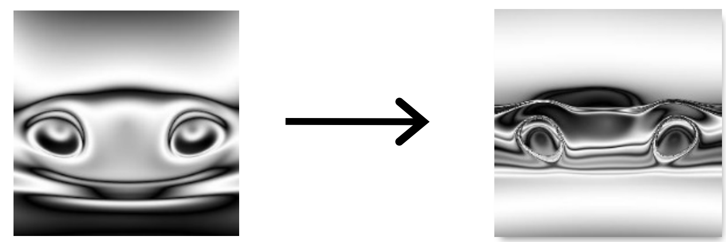
Stepping stone to the Teapot



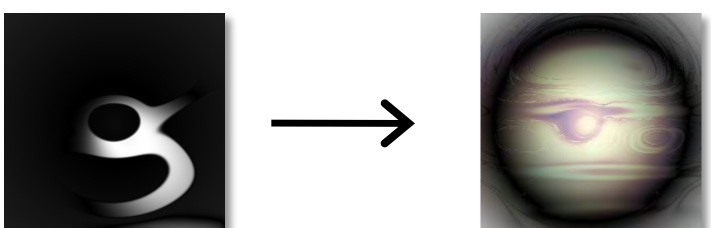
Stepping stone to the Butterfly



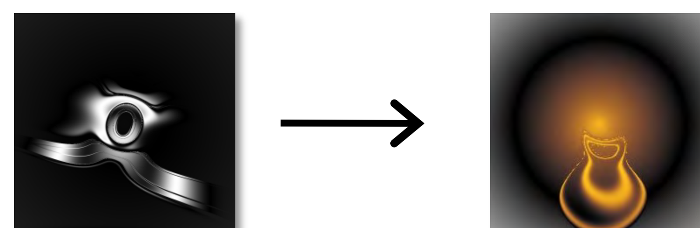
Stepping stone to the Skull



Stepping stone to the Penguin



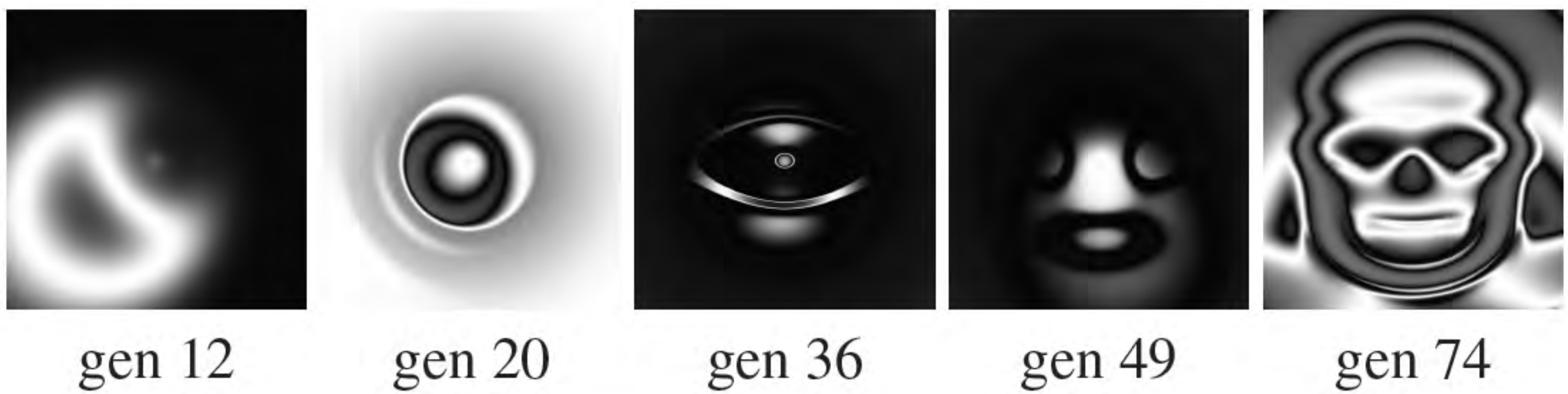
Stepping stone to Jupiter



Stepping stone to the Lamp

The stepping stones almost never resemble the final product!
Moral: You can only find things by not looking for them

Why?
Deception



(This insight is an inspiration for novelty search)

But without Humans, What Are the Necessary Conditions?

- What conditions are essential for open-endedness in general?
  - Hypotheses go back to Waddington (1969) and later Taylor (2012, 2015)
- Drawing on insights from population-based search, Soros and Staley (2014) propose our own
  - And that the system must generate new challenges *as well as* new ways to solve them

Proposed Necessary Conditions (Soros and Stanley 2014)

1. A non-trivial minimal criterion (MC) to proliferate
2. Individuals create new novel opportunities to satisfy the MC
3. Individual decide for themselves with what or whom to interact
4. Ability to increase the size of the representation (increasing information)

Proposed Necessary Conditions (Soros and Stanley 2014)

1. A non-trivial minimal criterion (MC) to proliferate
2. Individuals create new novel opportunities to satisfy the MC
3. Individual decide for themselves with what or whom to interact ← Coevolution, aka self-play
4. Ability to increase the size of the representation (increasing information)

Coevolution and Self-Play

- Interaction among learning agents (or changing components) intrinsically creates new challenges

Popovici, Elena, Anthony Bucci, R. Paul Wiegand, and Edwin D. De Jong. "Coevolutionary principles." *Handbook of natural computing* (2012): 987-1033.

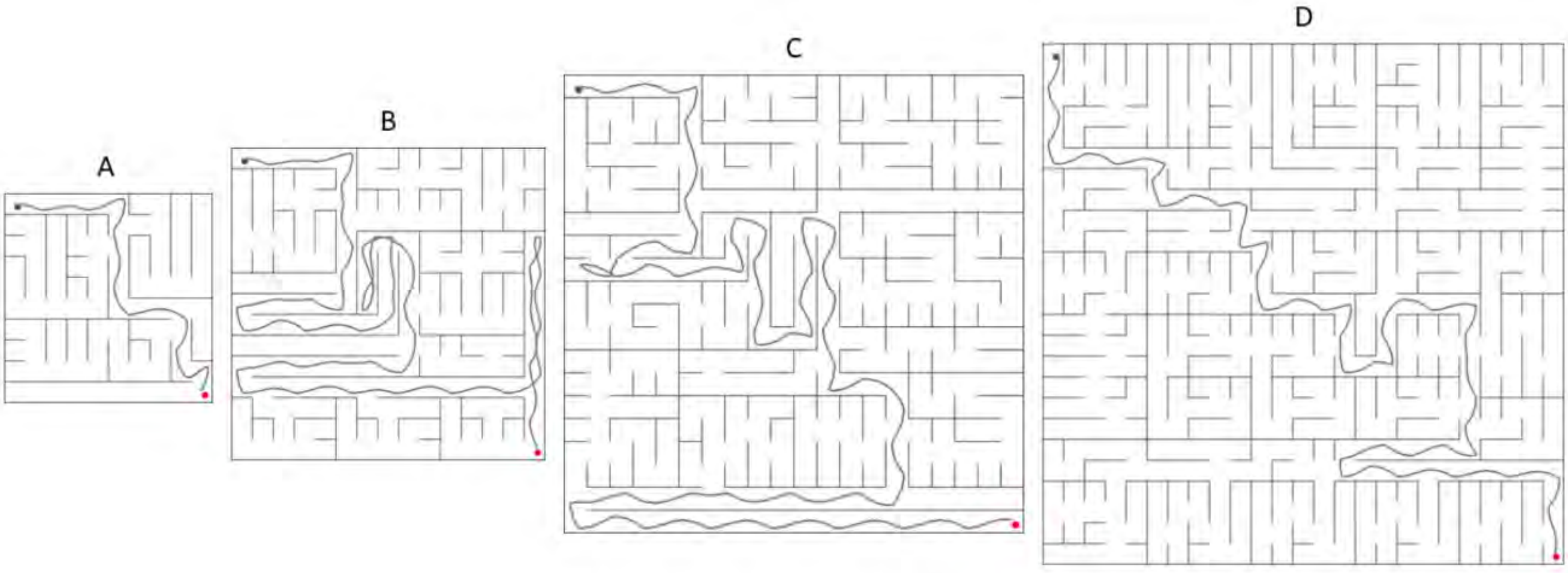
- Long studied in the field of *coevolution*
  – Competitive, cooperative, test-based
  – Drawing on game theory (Pareto-coevolution)
- More recently called *self-play*
  – OpenAI Five on Dota, AlphaGo and AlphaStar on Go and Starcraft, etc.

Conditions+Coevolution Eventually Leads to *Minimal Criterion Coevolution* (MCC) (Brant and Stanley 2017)

- Abstract the necessary conditions outside of alife worlds
  - Minimal criterion, self-generating opportunities
  - Leverage two-population coevolution to be domain-general
- First test: Mazes and maze solvers

Brant, Jonathan C., and Kenneth O. Stanley. "Minimal Criterion Coevolution: A New Approach to Open-Ended Search." Proceedings of the Genetic and Evolutionary Computation Conference (GECCO). 2017.

Single Run MCC Results – Mazes and Solutions of Unbounded Increasing Complexity



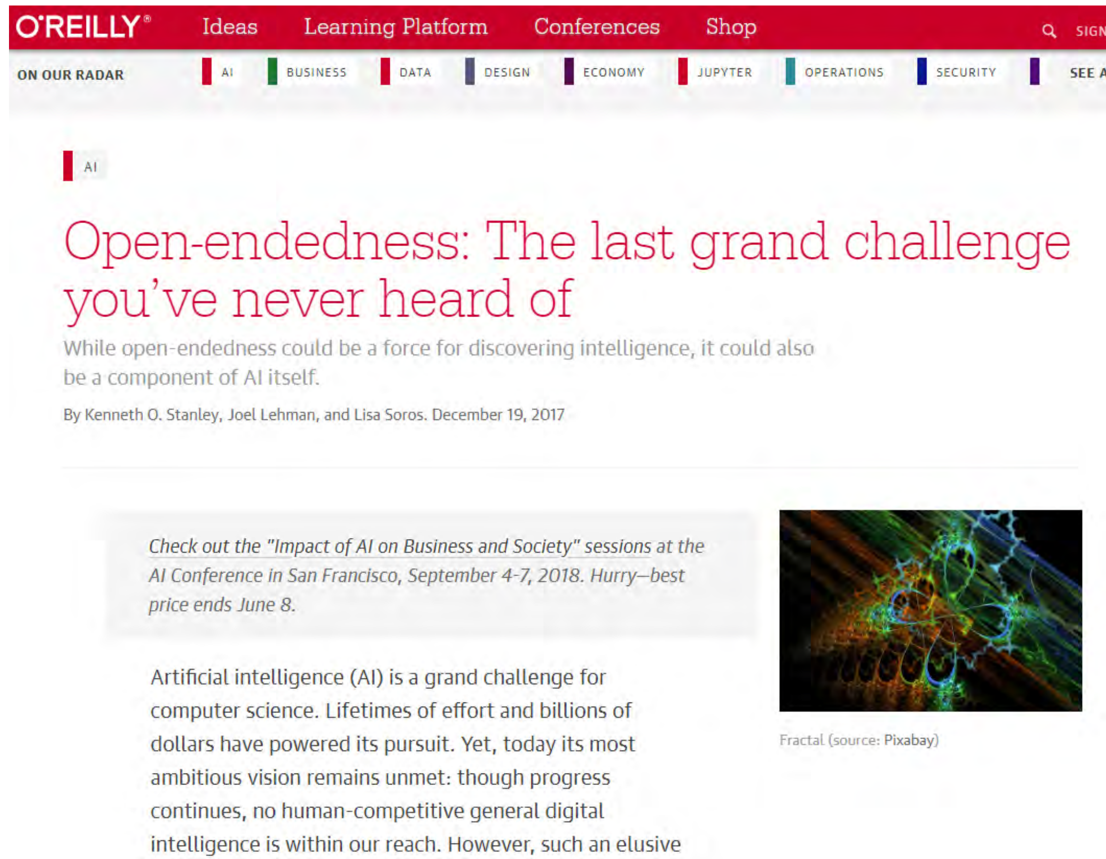
And, most recently, POET…

Open-Endedness: We're not Finished

- Field is just beginning; many challenges remain
  - Generating endless high-quality, diverse, and *interesting* artifacts remains a challenge
  - Killer applications remain critical for motivation
  - The measurement of success remains controversial and open
- *Open-endedness is the power of creation*
  - All of living nature is its product in a single run
  - When will we harness this power?

A Place to Start

- Non-technical intro to field (2017): https://www.oreilly.com/ideas/open-endedness-the-last-grand-challenge-youve-never-heard-of



O'REILLY® Ideas Learning Platform Conferences Shop SIGN IN

ON OUR RADAR AI BUSINESS DATA DESIGN ECONOMY JUPYTER OPERATIONS SECURITY SEE ALL

AI

Open-endedness: The last grand challenge you've never heard of

While open-endedness could be a force for discovering intelligence, it could also be a component of AI itself.

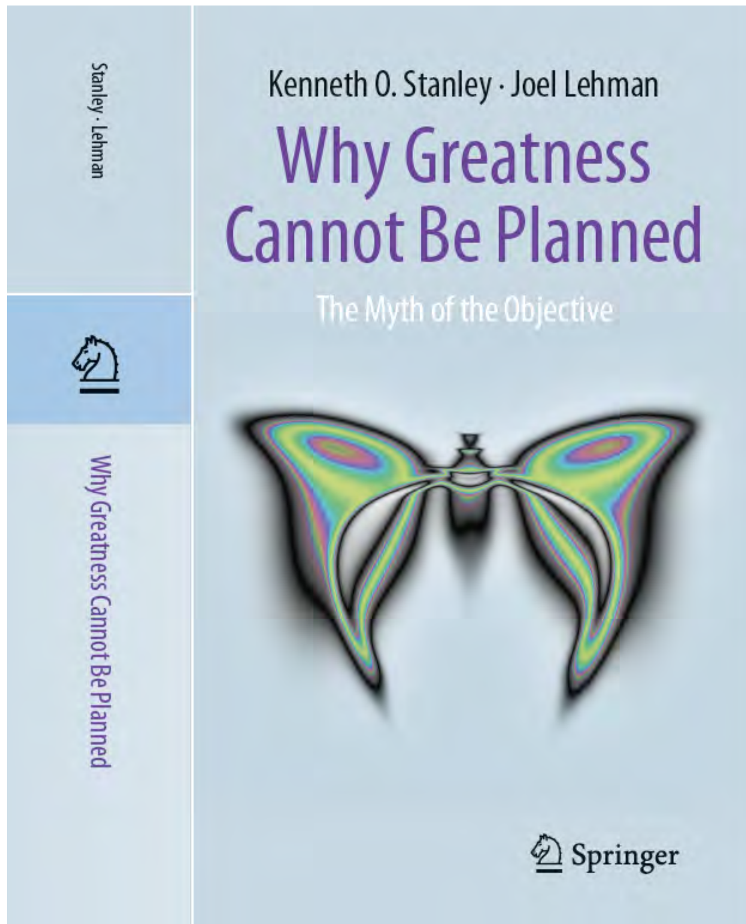
By Kenneth O. Stanley, Joel Lehman, and Lisa Soros. December 19, 2017

Check out the "Impact of AI on Business and Society" sessions at the AI Conference in San Francisco, September 4-7, 2018. Hurry—best price ends June 8.

Artificial intelligence (AI) is a grand challenge for computer science. Lifetimes of effort and billions of dollars have powered its pursuit. Yet, today its most ambitious vision remains unmet: though progress continues, no human-competitive general digital intelligence is within our reach. However, such an elusive

Fractal (source: Pixabay)

More Thoughts on Divergent Search



- Designing training environments is hard, but critical for progress
- Can machine learning algorithms generate their own training environments?

Paired Open-Ended Trailblazer (POET)

Rui Wang

Joel Lehman

Jeff Clune*

Ken Stanley*

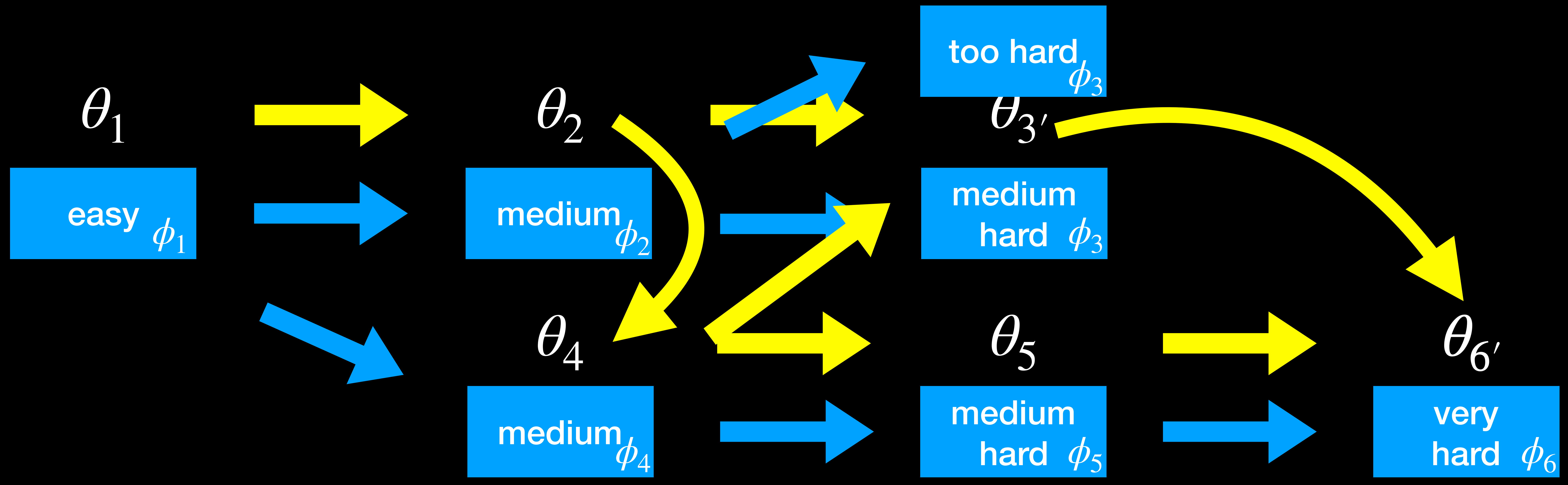
*Co-senior authors

2019

Automatically generates both challenges and solutions

Optimizes within niches & harnesses goal switching

POET



direct optimization fails

direct-path curriculum fails


298
207
304
311
349
309

POET

- Quality Diversity++
  - seeks the best agent for each niche
  - also generates niches
- Open-ended?
  - Definitely a step closer
  - Currently limited by
    - physics simulator
    - environmental encoding
  - Fully expressive environmental encoding: Generative Teaching Networks
    - ICML AutoML Workshop this Friday. Petroski-Such et al.

Automatically Generating Environments & Solutions

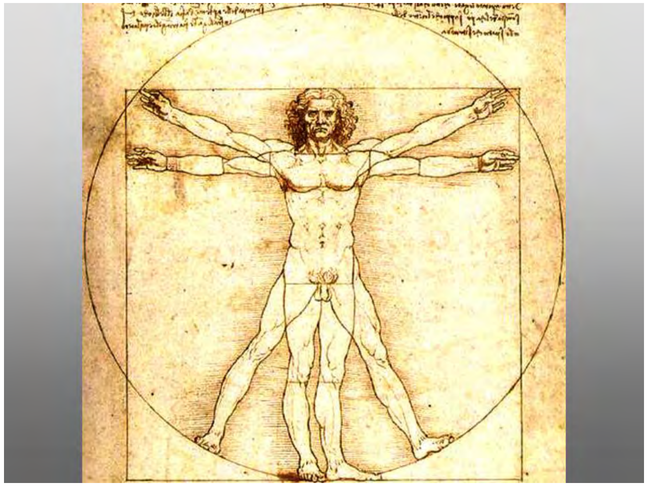
- Invents a curriculum
  - manual attempts fail
  - oven very counterintuitive (e.g. harder tasks help solve simpler ones)
- Endlessly innovates
- May be the only way to
  - solve ambitious problems
  - discover the full gamut of what is possible
- Captures spirit of open-ended engines of innovation
  - Natural evolution
  - Cultural evolution (science, technology, art)

Indirect Encoding: Representation in the Pursuit of Diversity

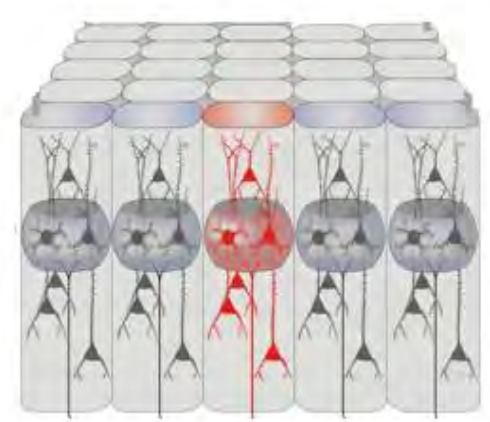
- When search is divergent…
  - The likely trajectories through the space of designs become important
- Regularities should be possible to discover, and to preserve
- But regularity should also be flexible and allow exceptions

Therefore, Indirect Encoding

- Indirect encoding: “Genes” do not map directly to units of structure in phenotype
- Genetic material can be reused
- Development from DNA as *inspiration*



Symmetry



Repetition

Repetition with variation

Historical Precedent

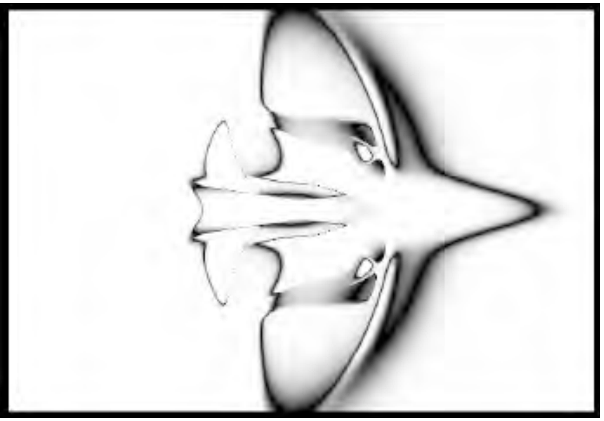
- Turing (1952) was interested in morphogenesis
  - Experimented with reaction-diffusion equations in pattern generation
- Lindenmayer (1968) investigated plant growth
  - Developed L-systems, a grammatical rewrite system that abstracts how plants develop
- A long history of encodings

Stanley, Kenneth O., and Risto Miikkulainen. "A taxonomy for artificial embryogeny." *Artificial Life* 9.2 (2003): 93-130.

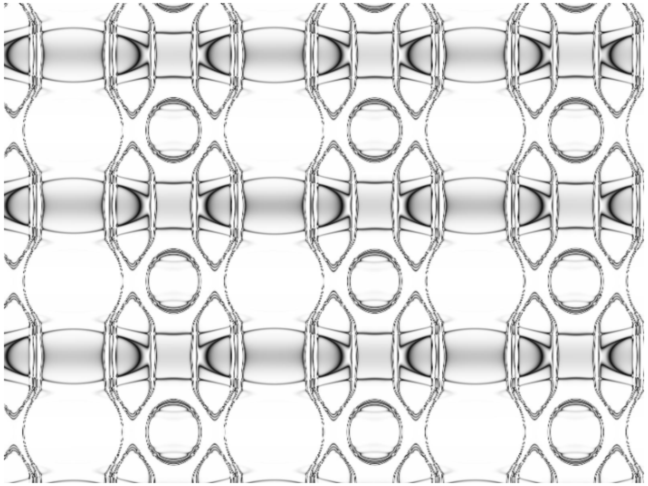
Lindenmayer, A. (1968). Mathematical models for cellular interaction in development: Parts I and II. Journal of Theoretical Biology, 18, 280–299, 300–315.
Turing, A. (1952). The chemical basis of morphogenesis. *Philosophical Transactions of the Royal Society B, 237*, 37–72.

High-Level Abstraction: Compositional Pattern Producing Networks (CPPNs)

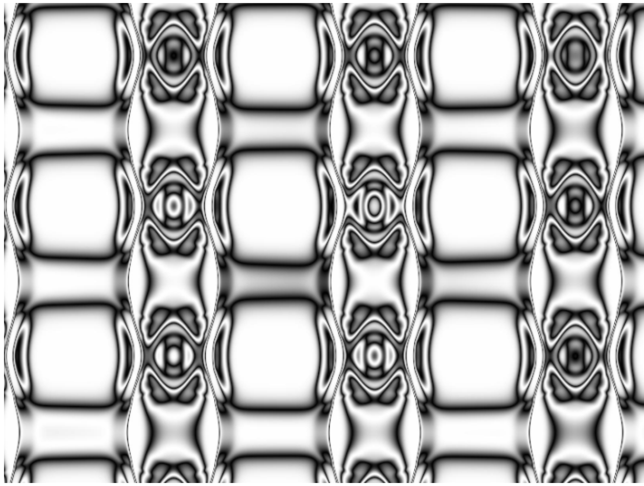
- IE suited to NNs designed to abstract how embryos are encoded through DNA (Stanley 2007)



Symmetry



Repetition

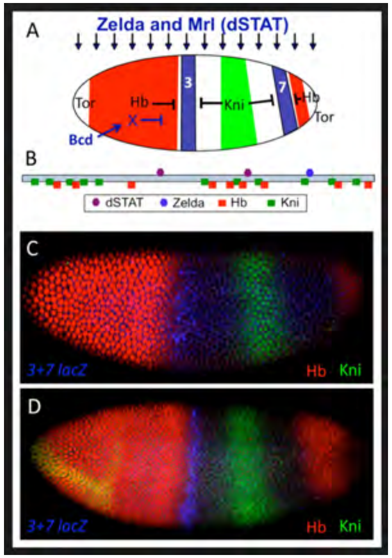


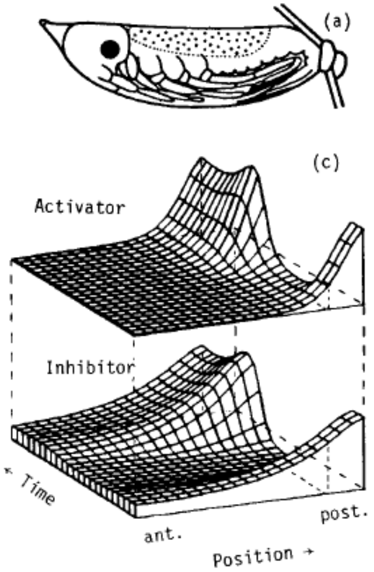
Repetition with variation

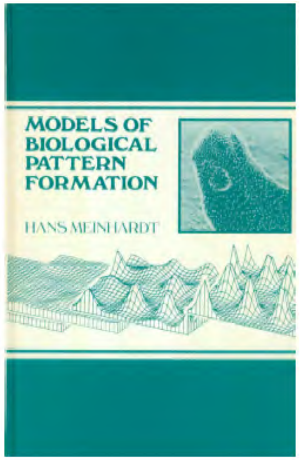
Kenneth O. Stanley.
Compositional Pattern Producing Networks: A Novel Abstraction of Development In: Genetic Programming and Evolvable Machines Special Issue on Developmental Systems 8(2): 131-162. New York, NY: Springer, 2007

Insight: In Embryogeny, Cells Know Where They Are Through Chemical Gradients

- Therefore, they know who needs to do what, and where
- Because *where* is now defined
- Gradients form a *coordinate frame*



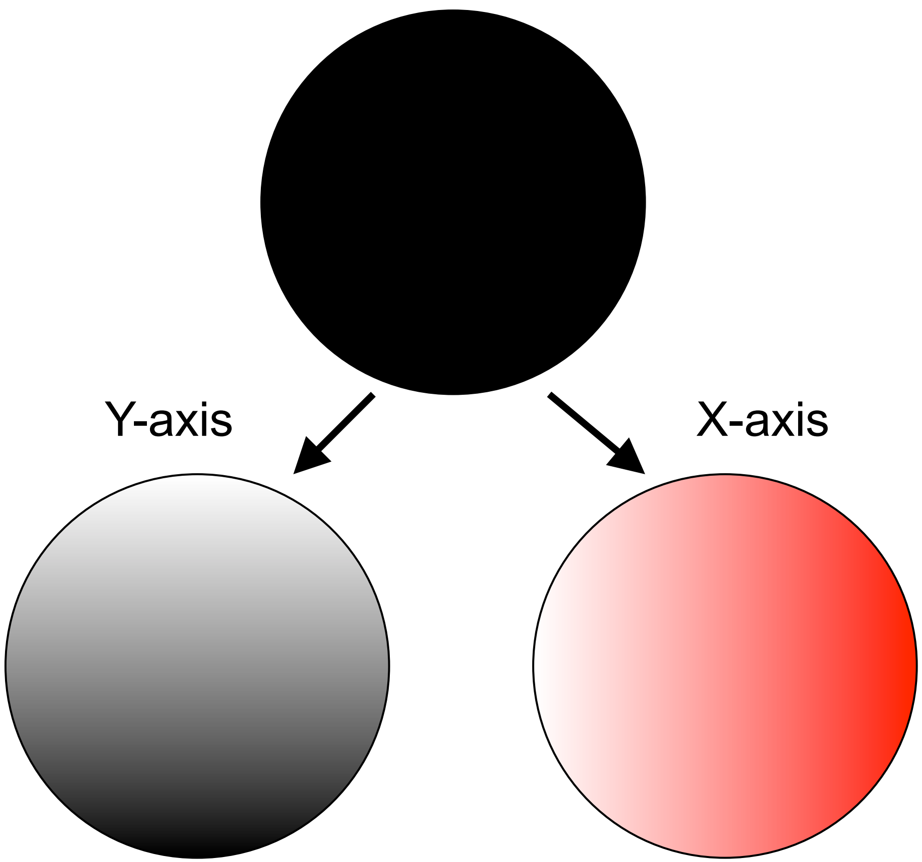




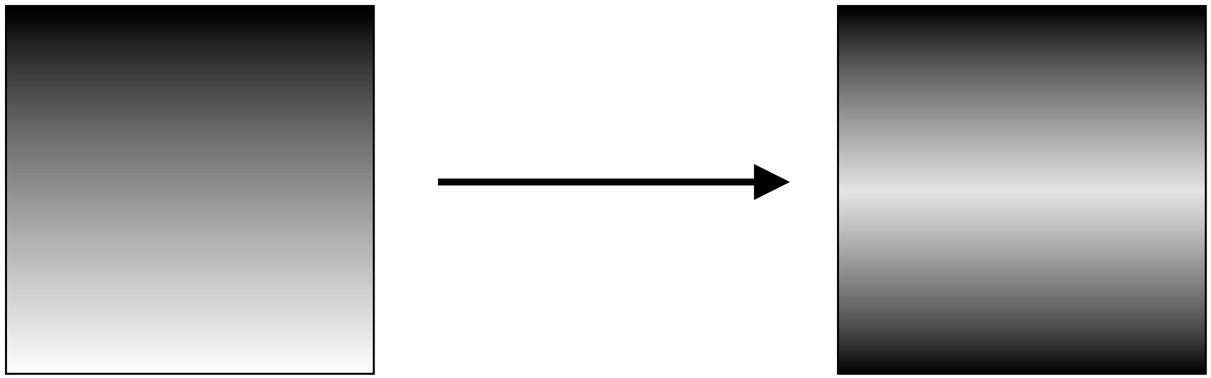
(1982)

Gradients Define Axes

- Chemical gradients tell which direction is which, which axis is which



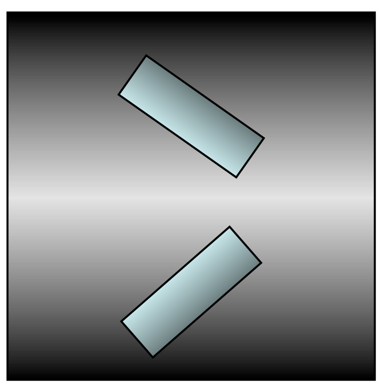
Higher Coordinate Frames are Functions of Lower Ones



$$f(y) = y \qquad\qquad g(y) = \left| f(y) \right|$$

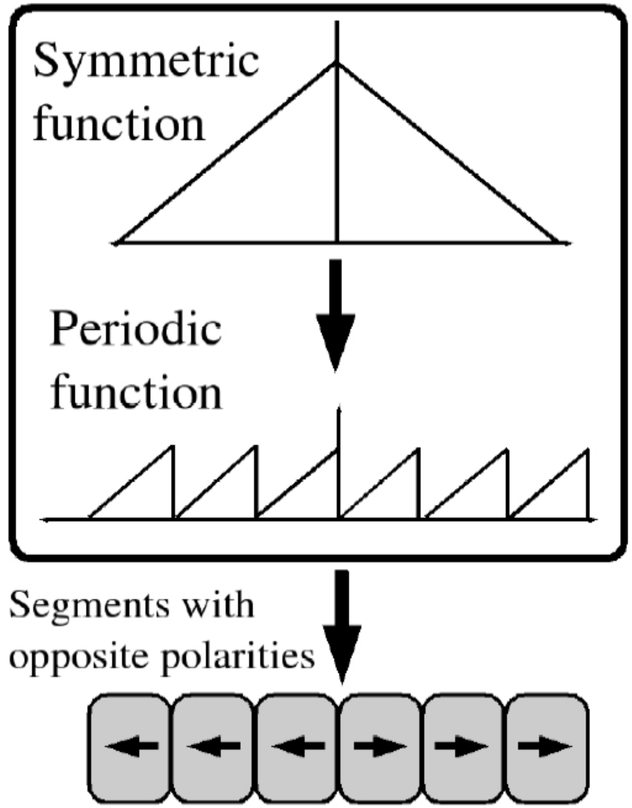
Using g and x as a coordinate space, we can get h:

Symmetry from a symmetric gradient



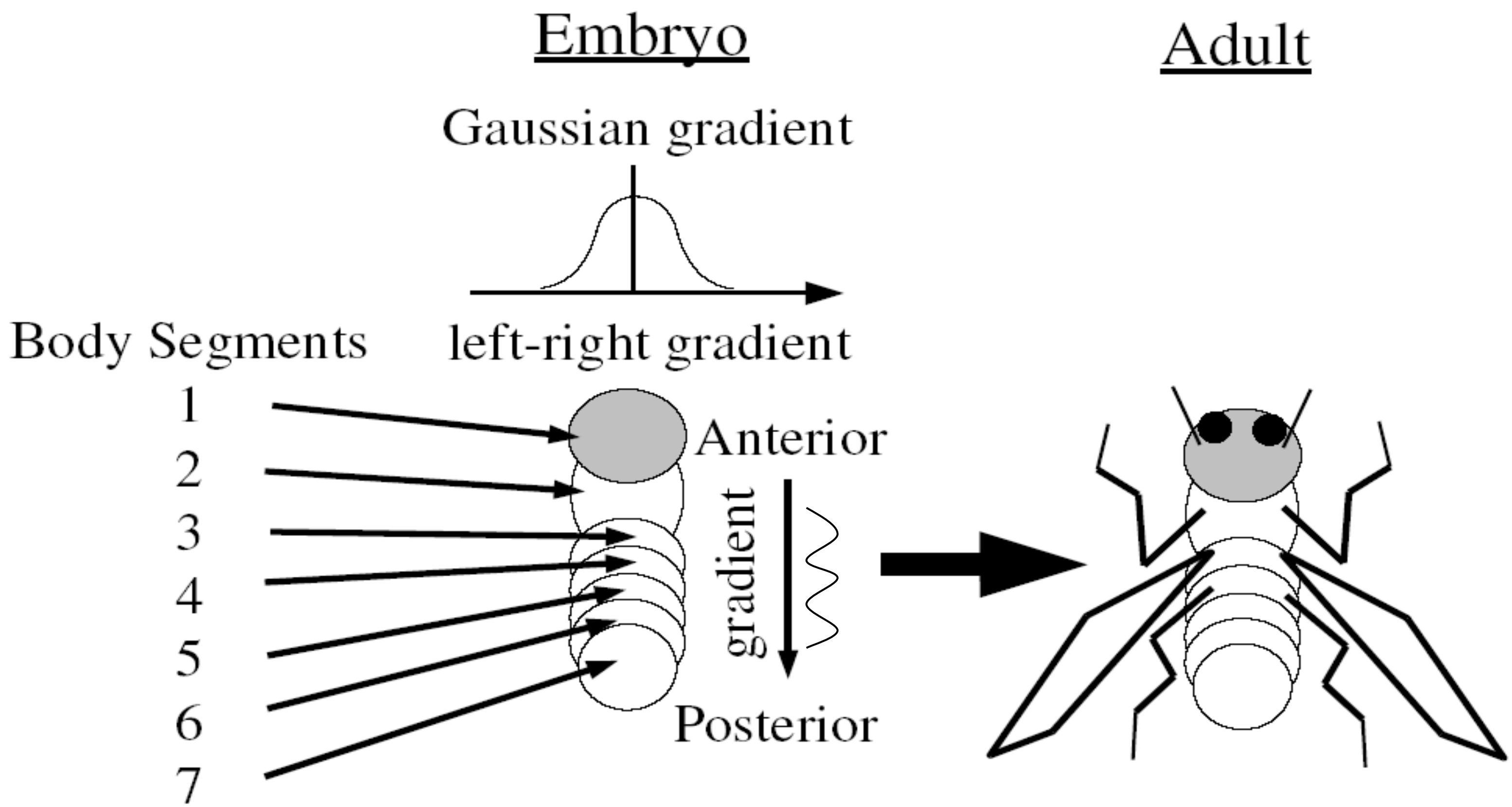
$$h(x,y) = func\left[x, g(y)\right]$$

Gradients Can Be Composed

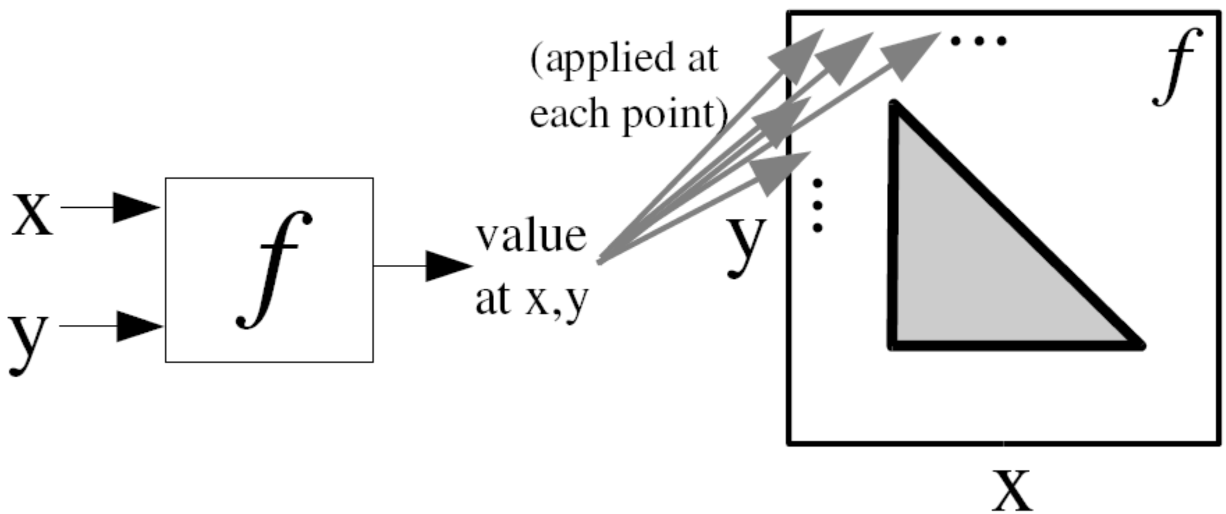


- Is there a general abstraction of composing gradients that we can evolve?

Gradients Define the Body Plan

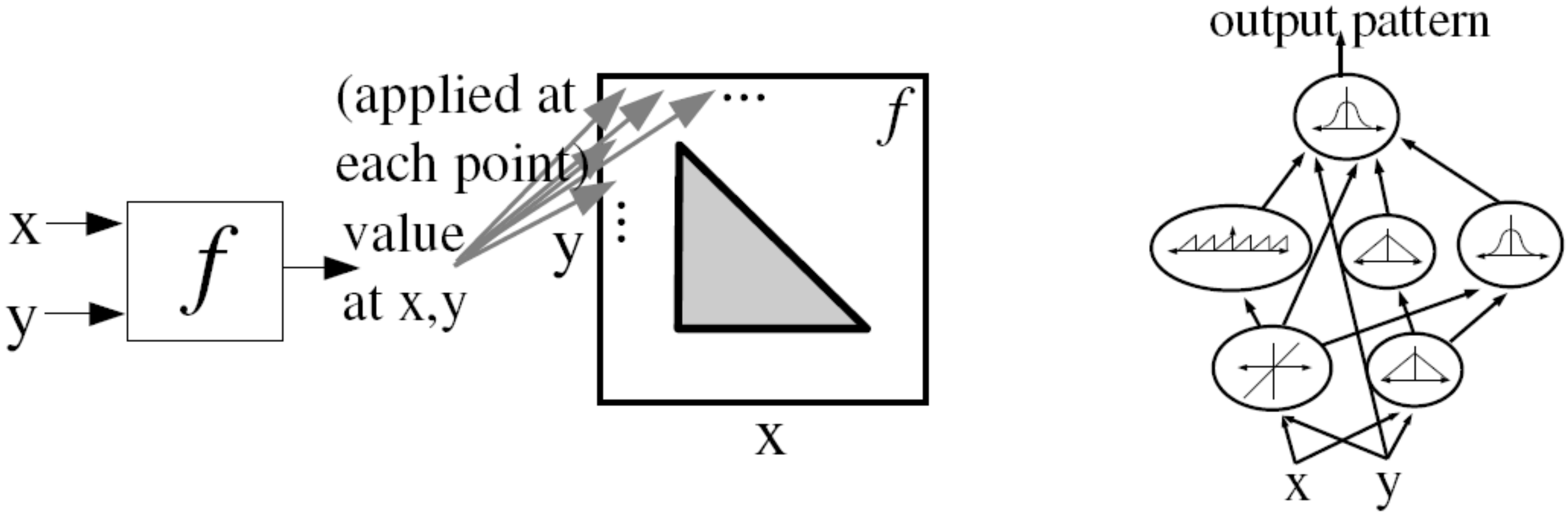


A Novel View: The Phenotype as a Function of Cartesian Space



- Coordinate frames are chemical gradients
- Function is applied at all points

Compositional Pattern Producing Networks (CPPNs)



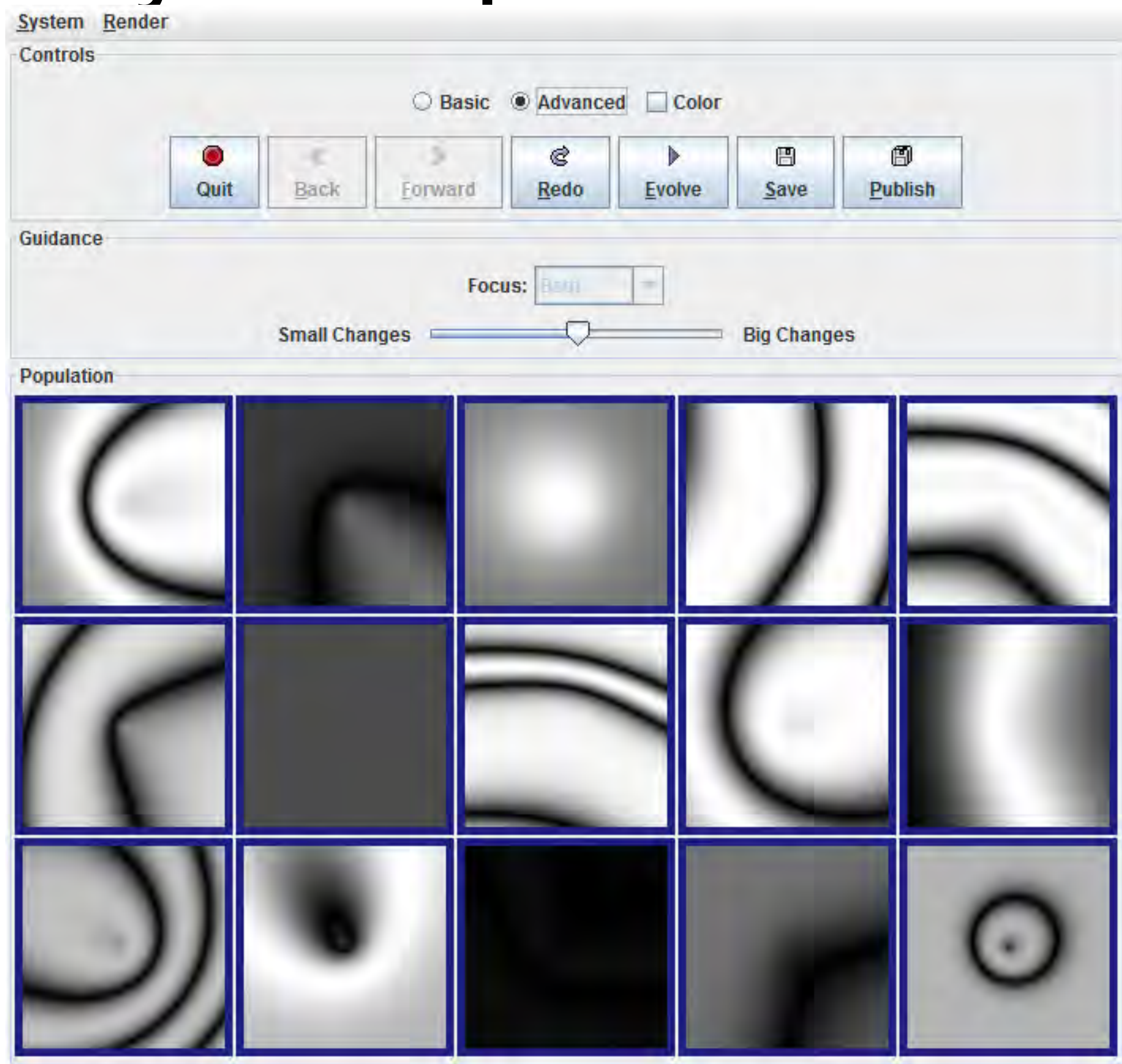
(a) Mapping (b) Composition

- A connected-graph abstraction of the order of and relationship between developmental events (no growth!)

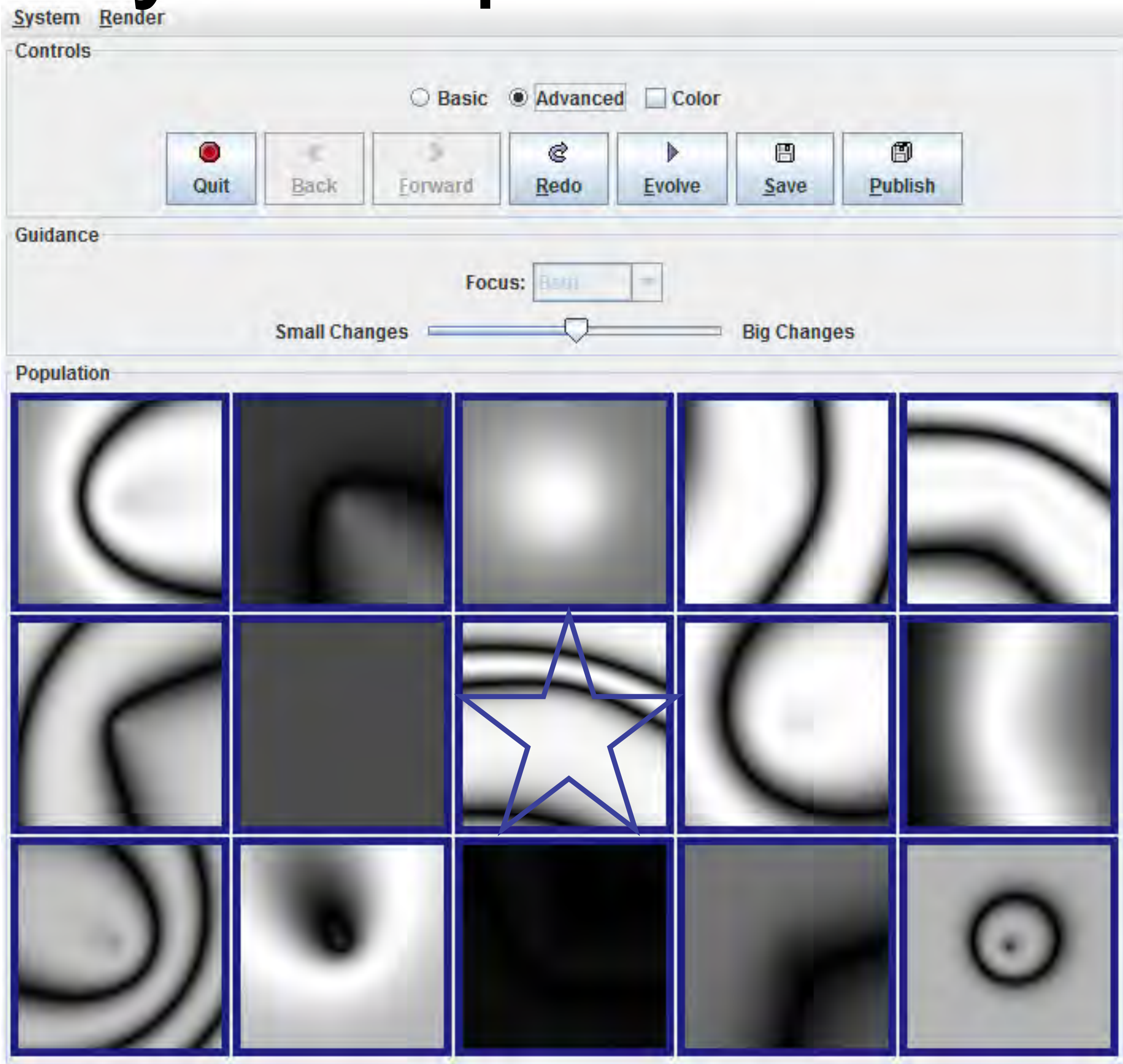
Searching Over CPPNs

- Method (for now): NEAT (Neuroevolution of Augmenting Topologies)
  - Evolves NNs of increasing complexity
  - Speciation for diversity
- Why evolve CPPNs with NEAT?
  - Increasing complexity allows for elaboration on existing patterns

Interactive Evolution: A Way to Explore Encoding

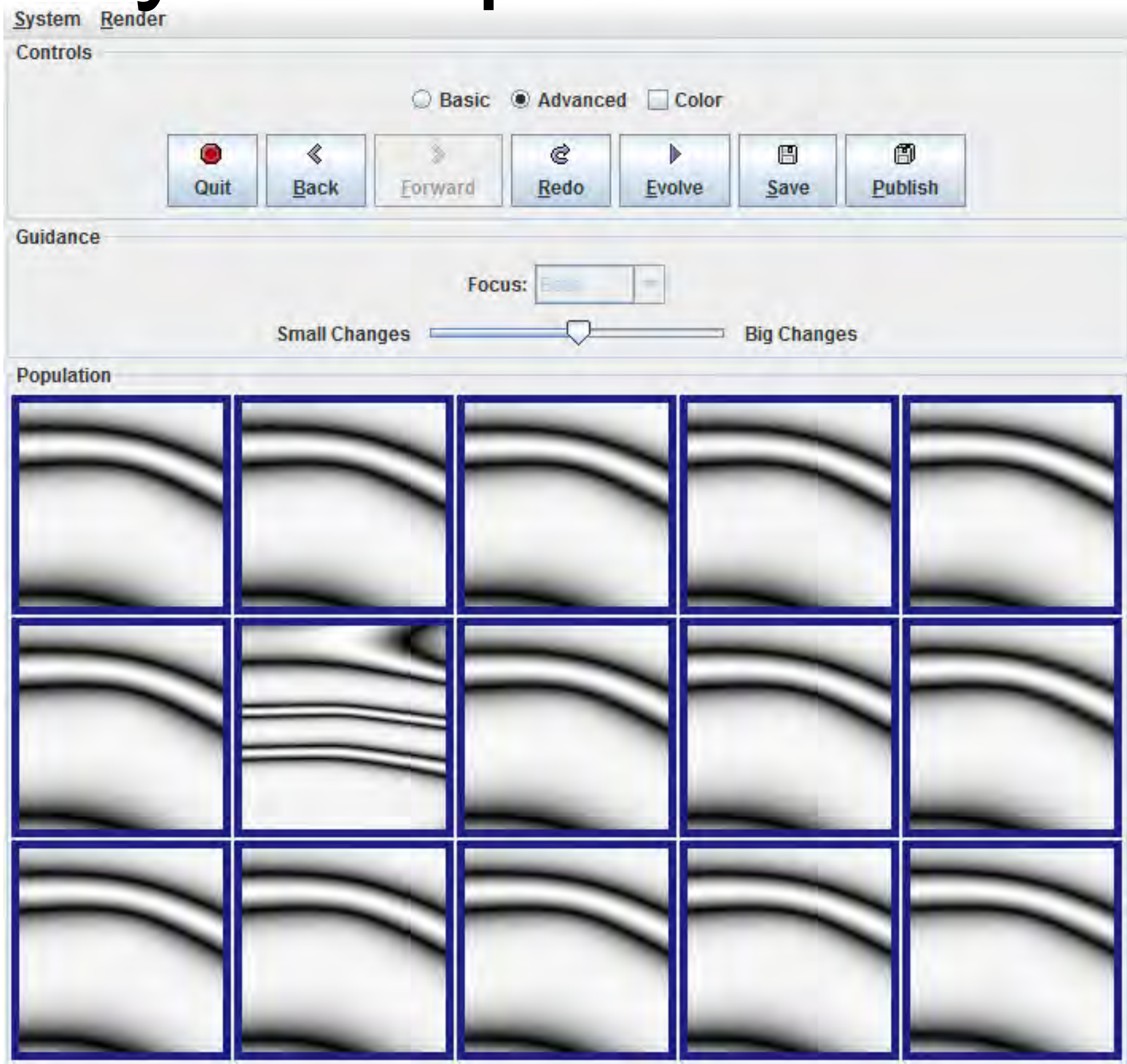


Interactive Evolution: A Way to Explore Encoding

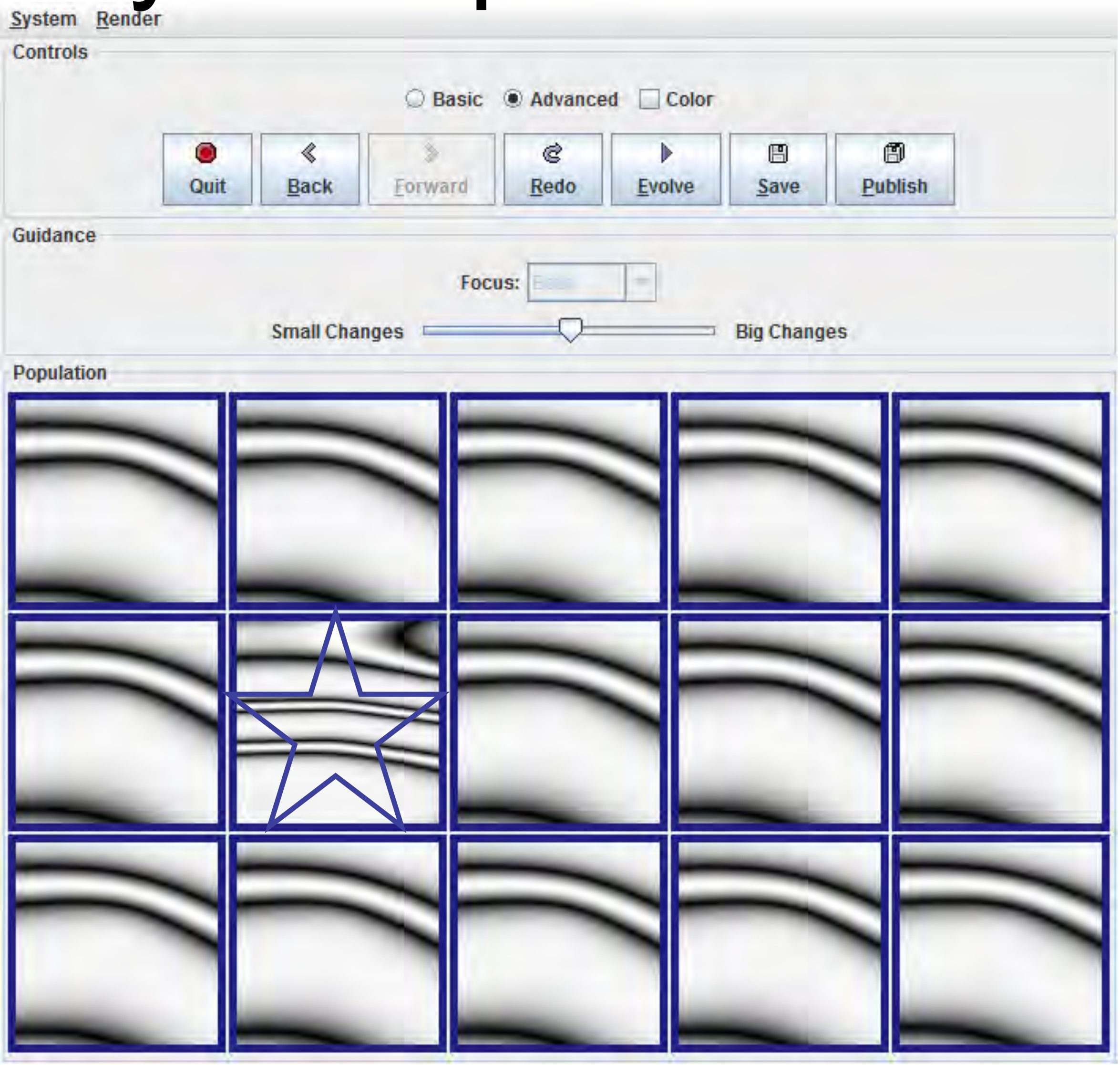


Interactive Evolution:
A Way to Explore Encoding

Parent

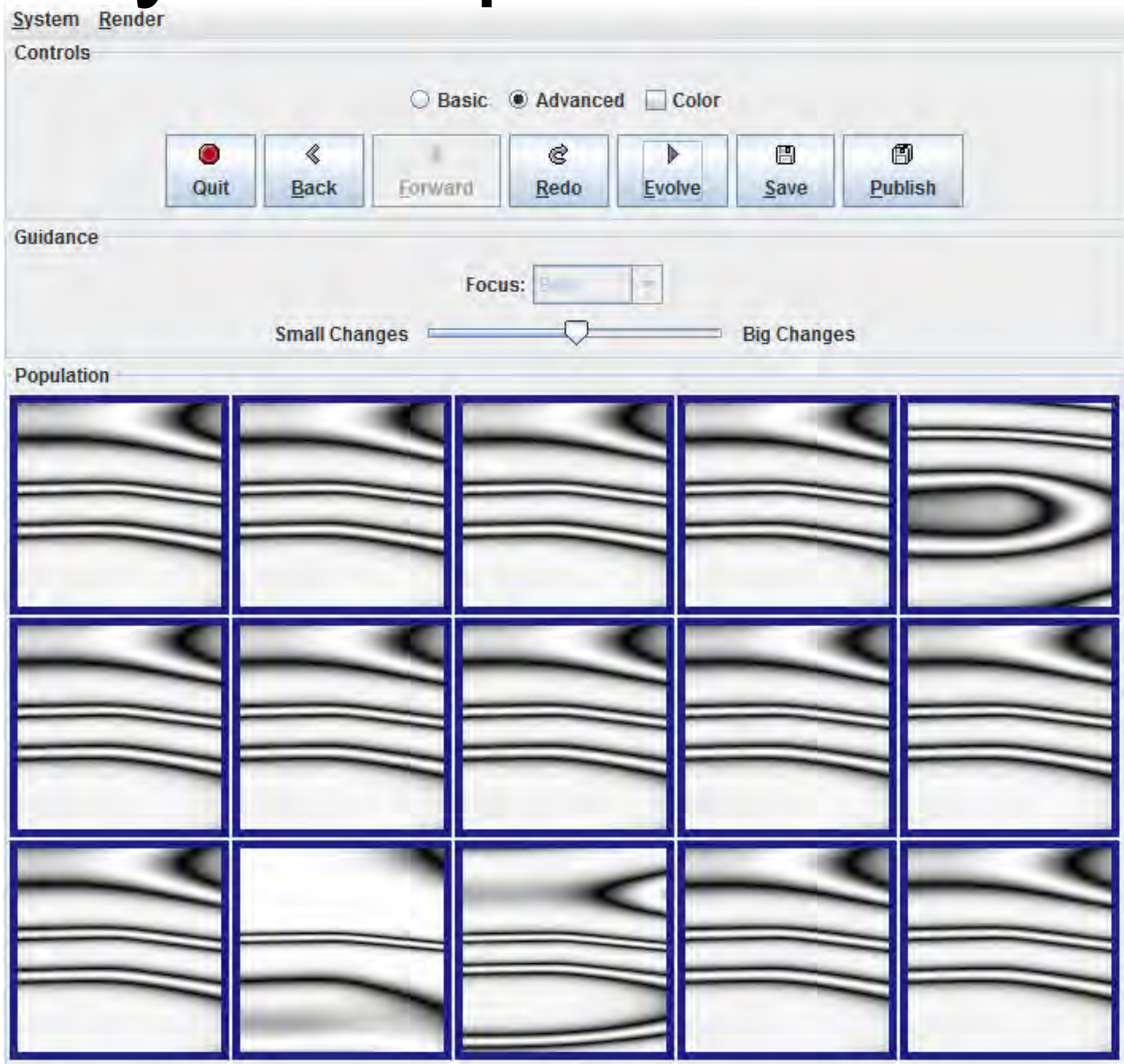


Interactive Evolution: A Way to Explore Encoding



Parent

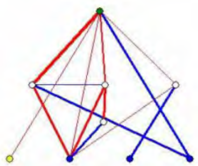
Interactive Evolution: A Way to Explore Encoding

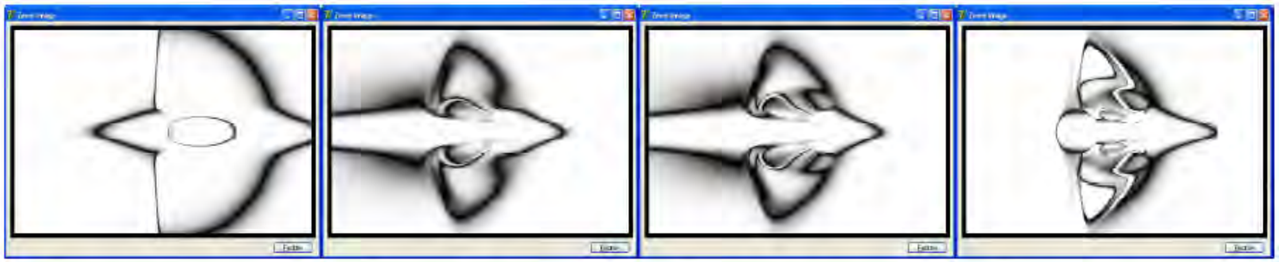


And so on…

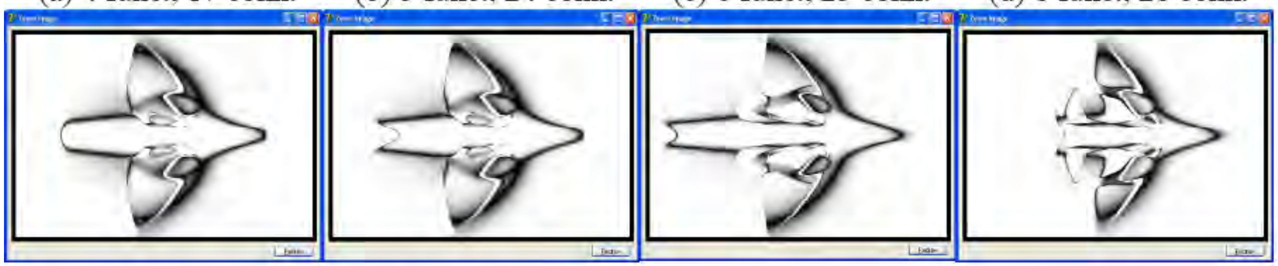
Parent

Evolutionary Elaboration with CPPNs

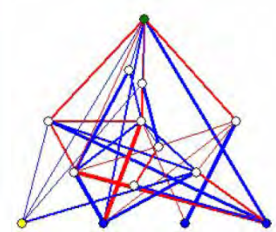




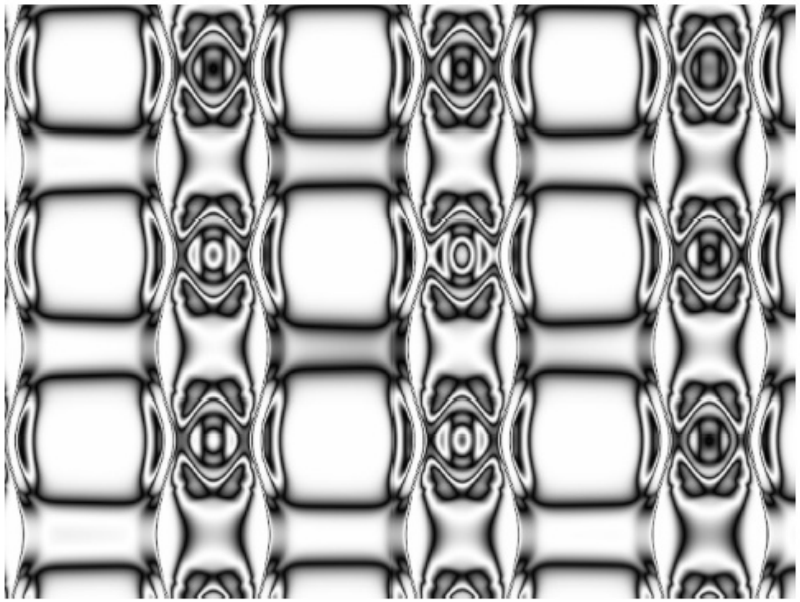
(a) 4 func., 17 conn. (b) 5 func., 24 conn. (c) 6 func., 25 conn. (d) 8 func., 28 conn.

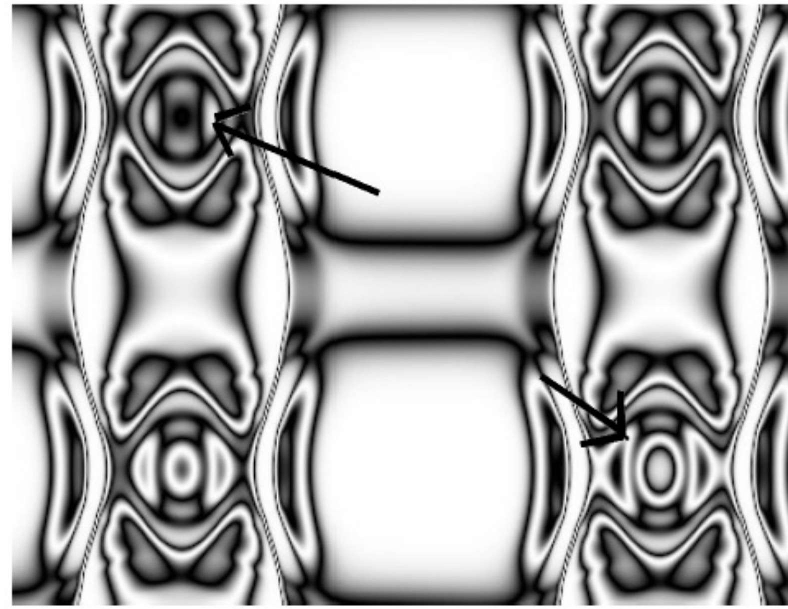


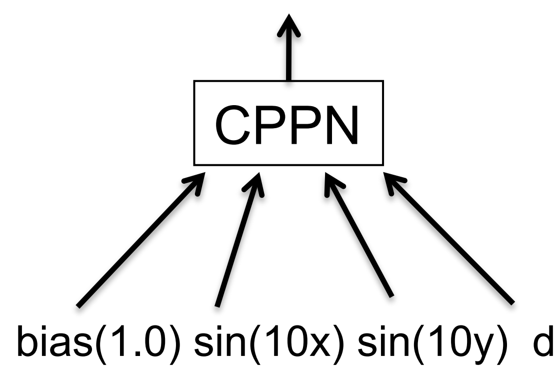
(e) 8 func., 30 conn. (f) 8 func., 31 conn. (g) 8 func., 32 conn. (h) 8 func., 34 conn.



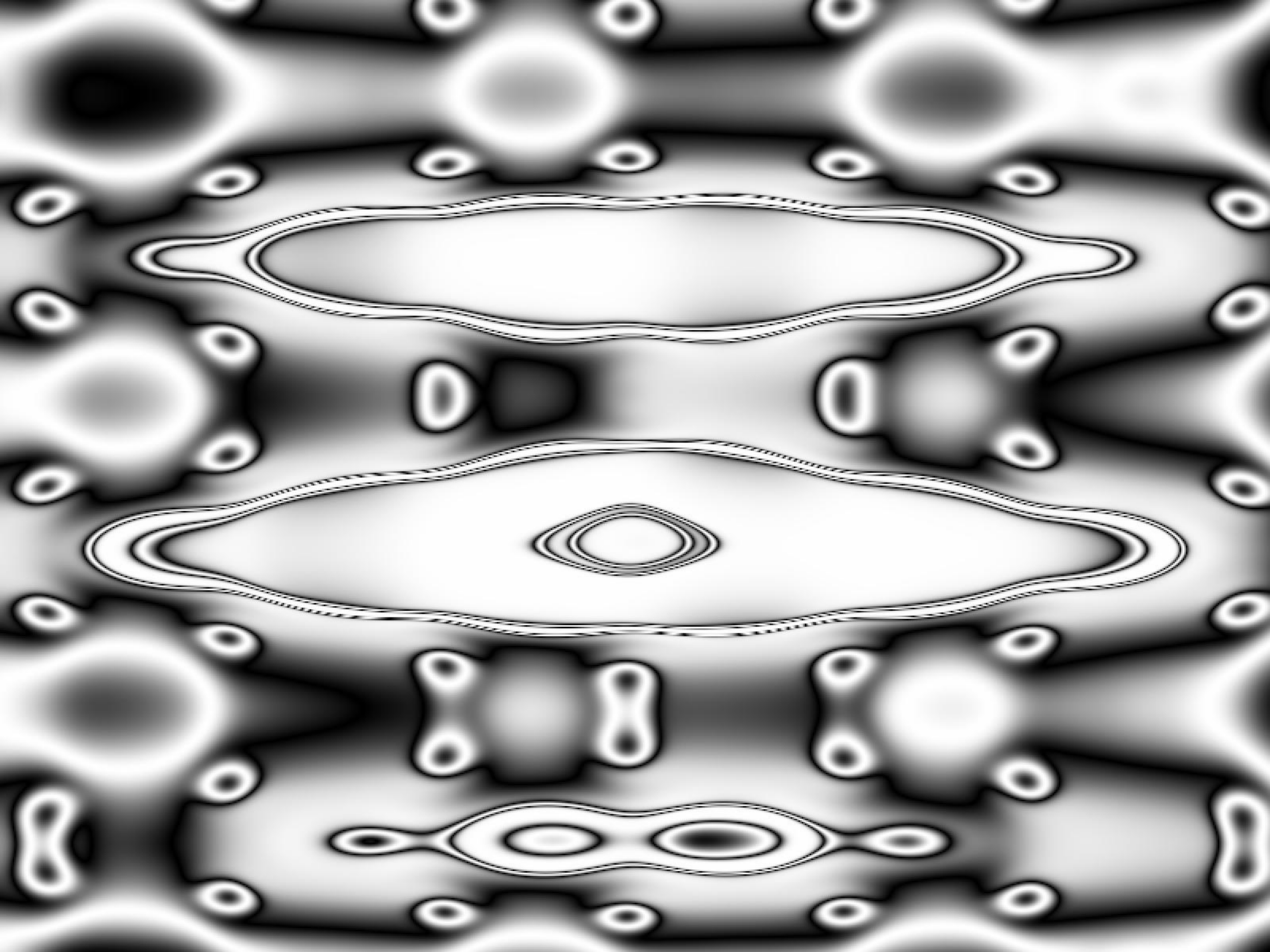
CPPNs:Repetition with Variation

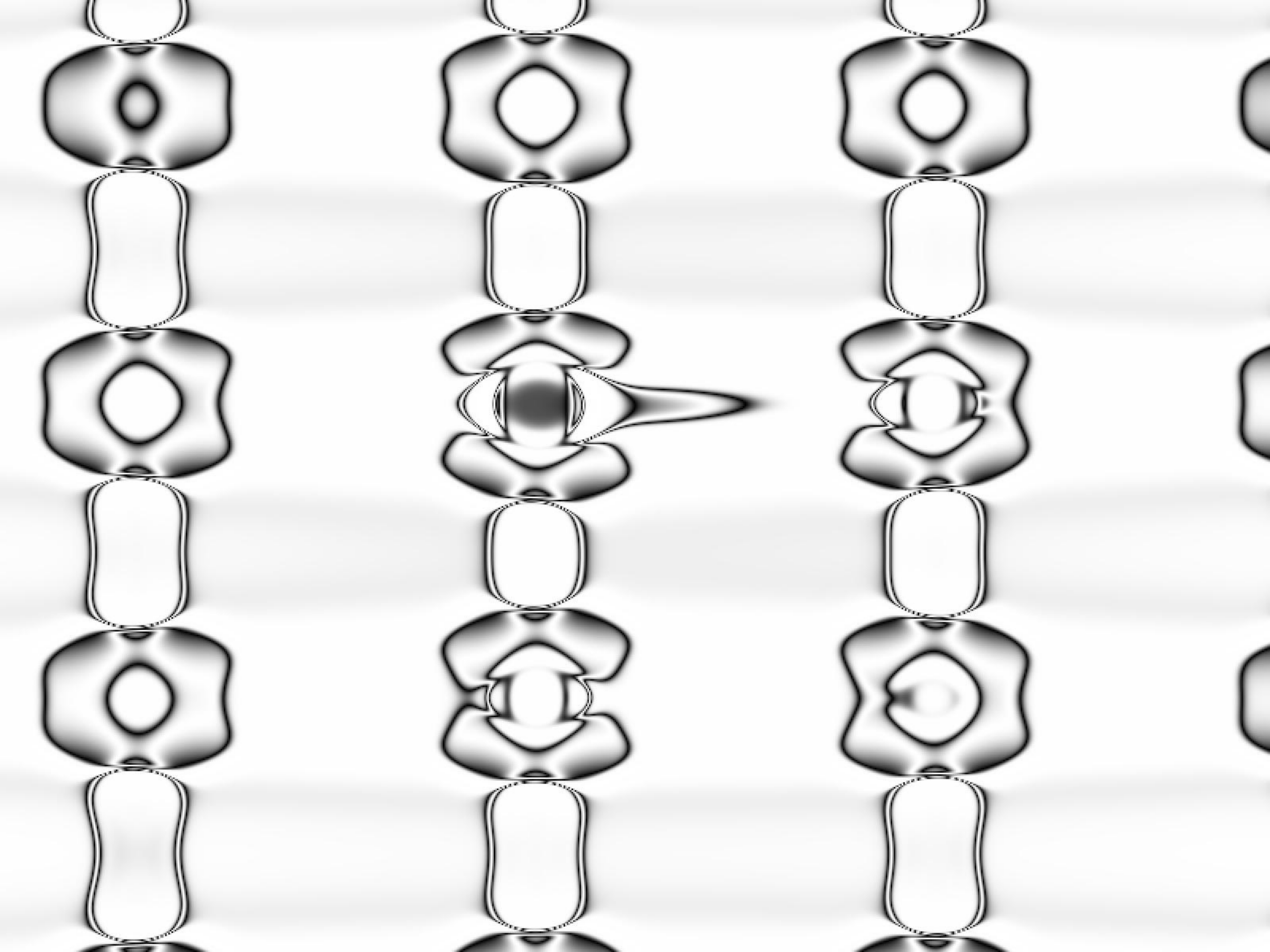


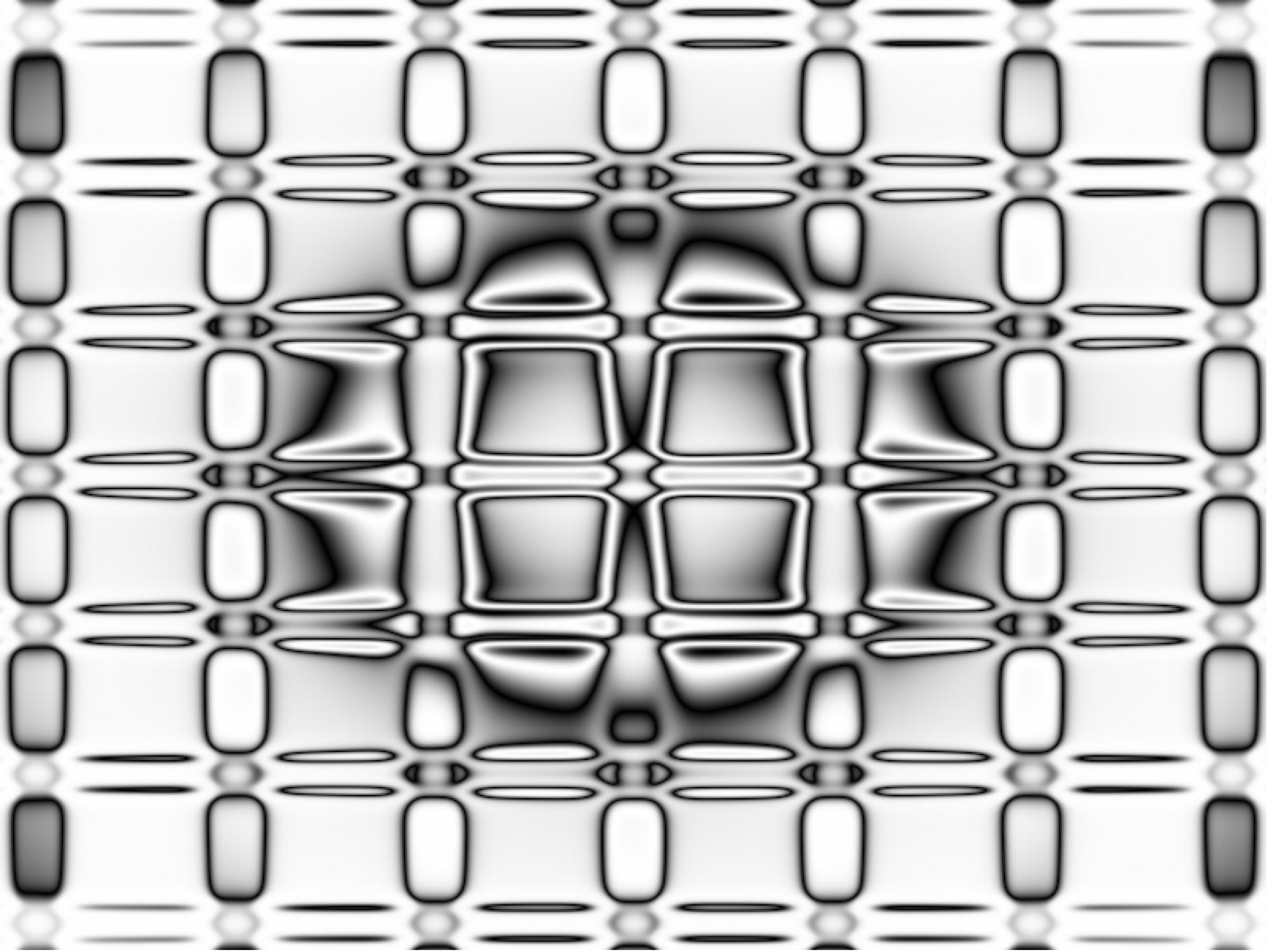


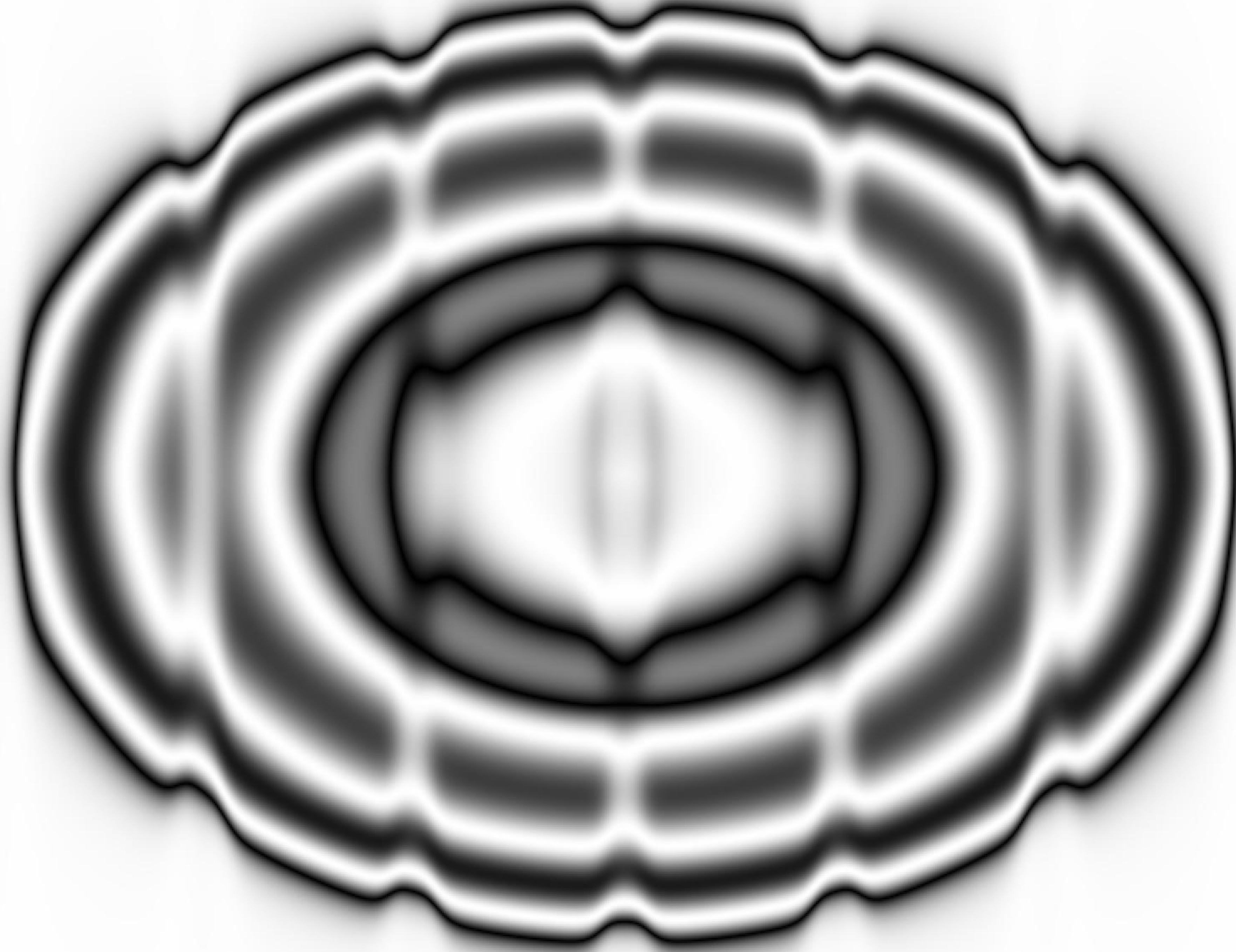


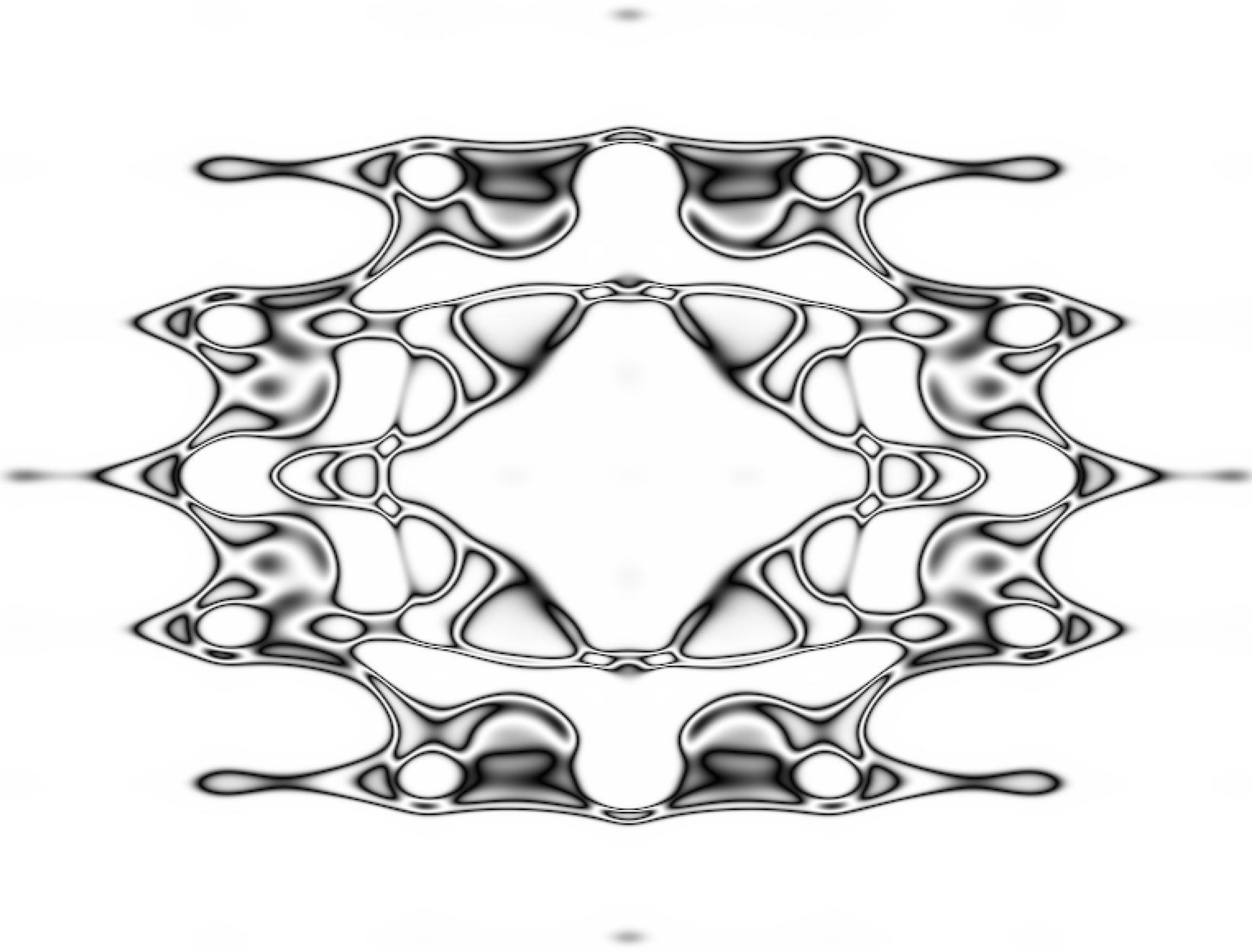
- Seen throughout nature
- A simple combination of periodic and absolute coordinate frames







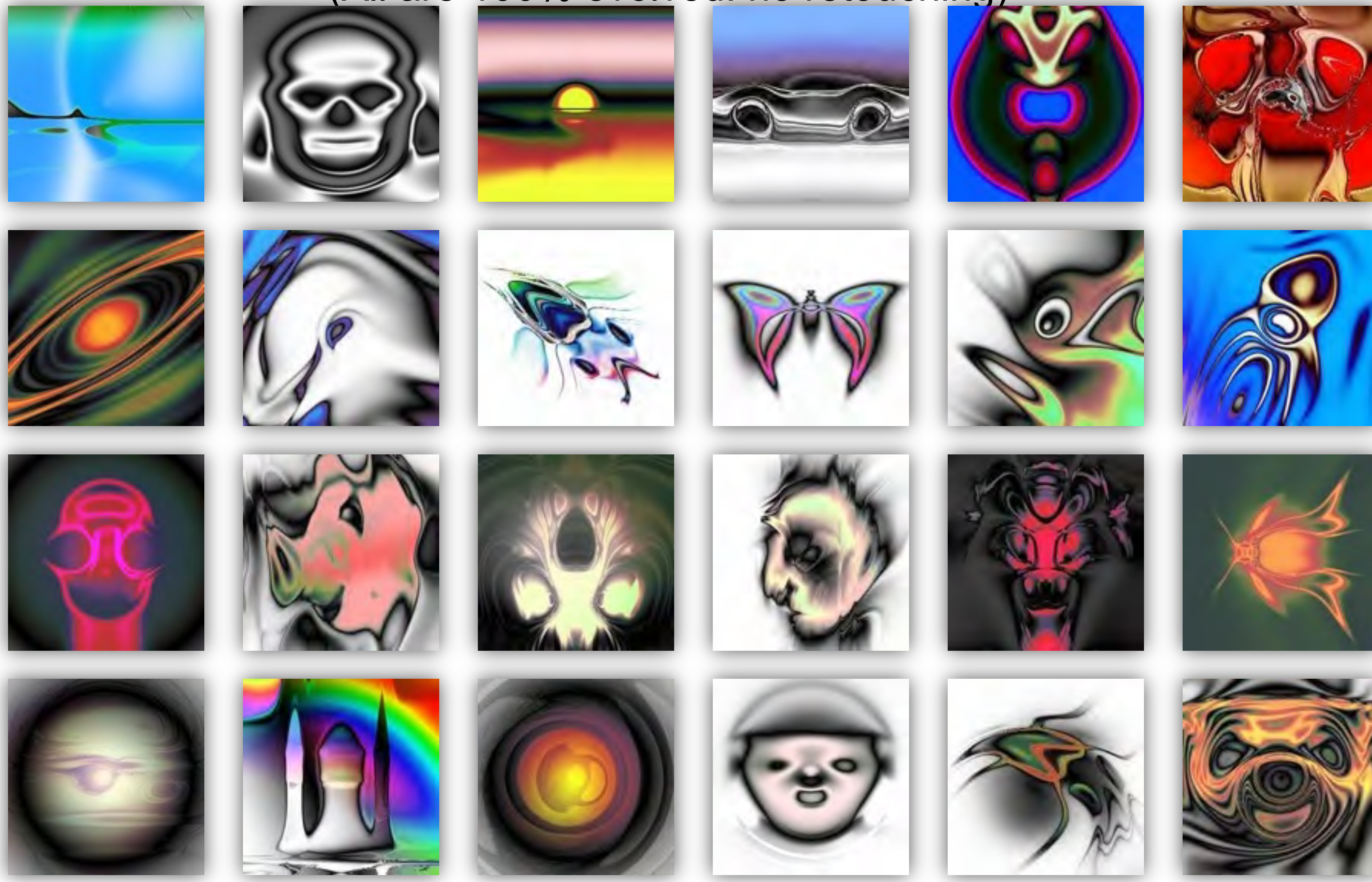




CPPN Patterns

From http://picbreeder.org

(All are 100% evolved: no retouching)



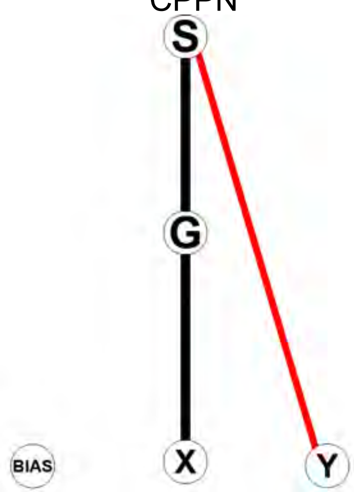
The Challenge

- CPPNs encode spatial patterns with regularities
- It would be nice if CPPNs could represent *networks* with similar regularities
- *How can CPPNs encode NNs?*

The Solution:
Hypercube-based NEAT (HyperNEAT)

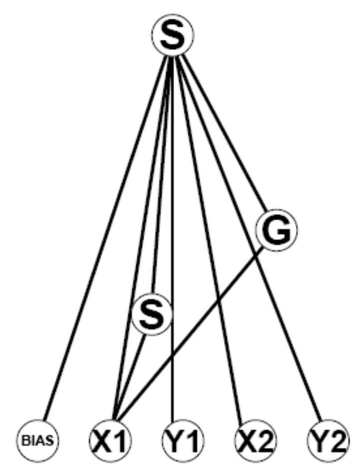
- Main insight: 2-D connections *isomorphic* to 4-D points
  - Nodes situated in 2 spatial dimensions (x,y)
  - Connections expressed with 4 spatial dim. (x_1,y_1,x_2,y_2)
- HyperNEAT extends 2-D CPPNs to 4-D (or 6-D)
  - CPPN encodes 4-D patterns (i.e. inside a hypercube)
    - 4-D patterns can express the same regularities as 2D patterns
    - 4-D patterns interpreted as connectivity patterns

CPPN

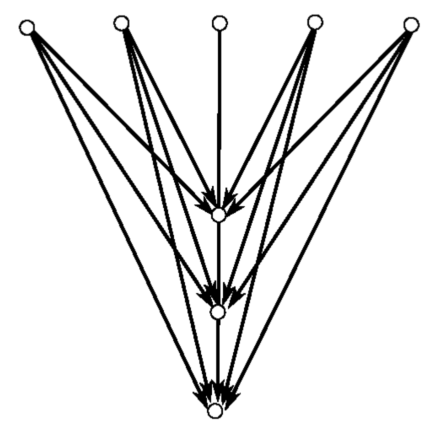


Output

CPPN

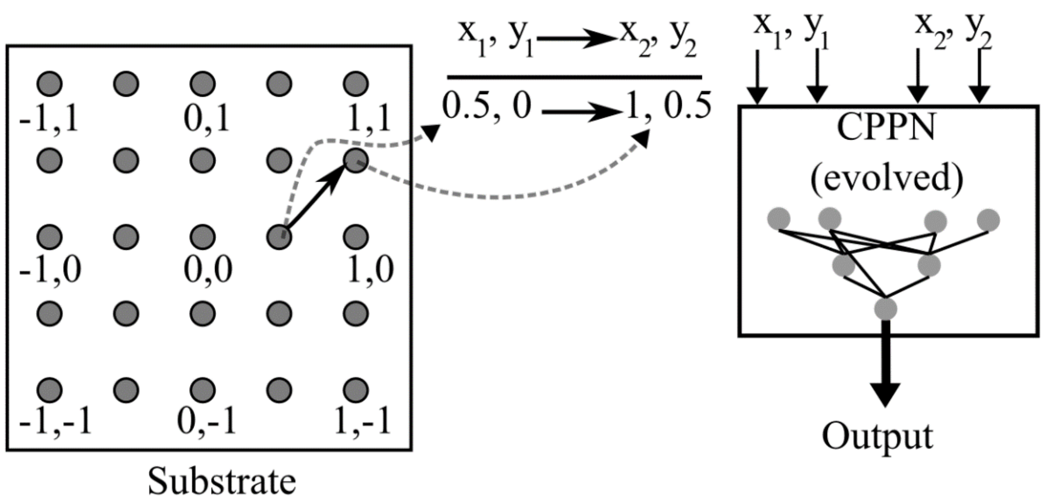


Output

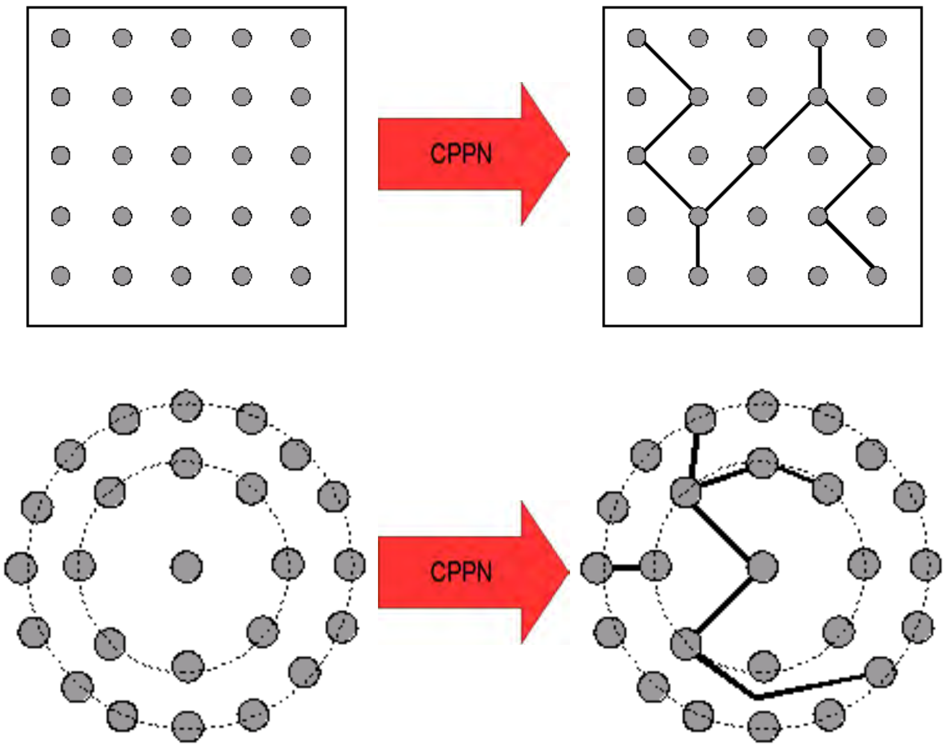


HyperNEAT

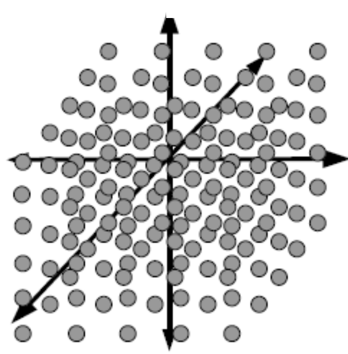
- 4-D CPPN
  - The network evolved by HyperNEAT
- Substrate
  - The NN encoded by the 4-D CPPN
  - A *function* of geometry, i.e. *sees the geometry*
  - Each connection is queried by the CPPN to retrieve a weight

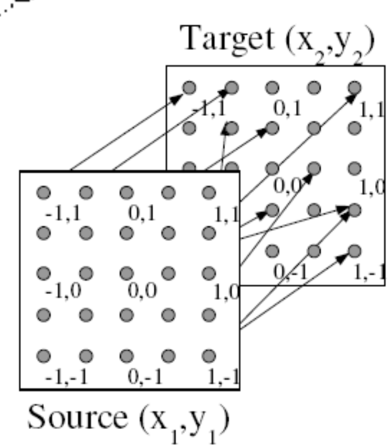


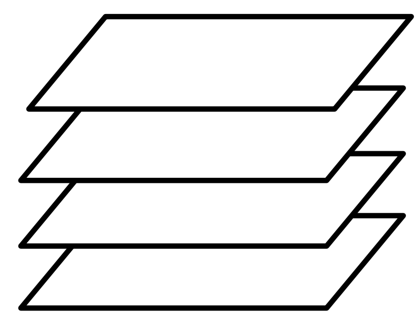
Substrates



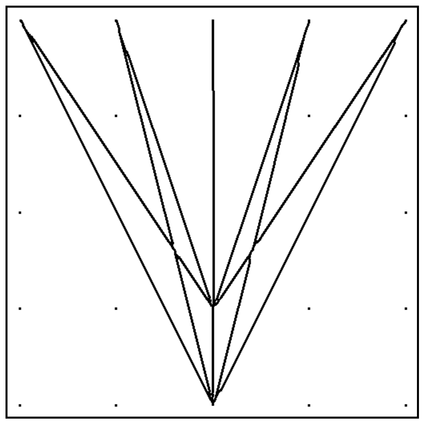
- Can be configured to best exploit problem geometry
  - Natural for many problems
- Input, Output, and Hidden nodes can be placed in any pattern
- Not restricted to 2-D



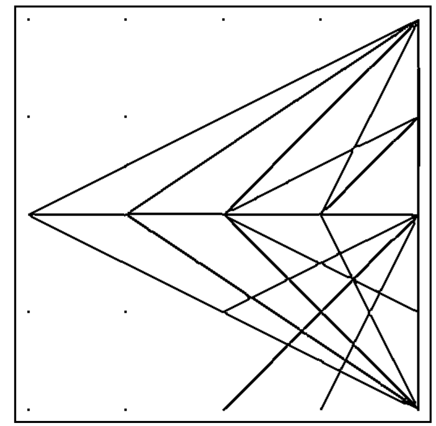




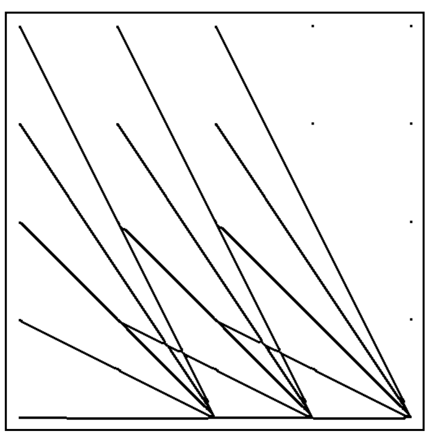
Fundamental Regularities Produced by 4-D CPPNs



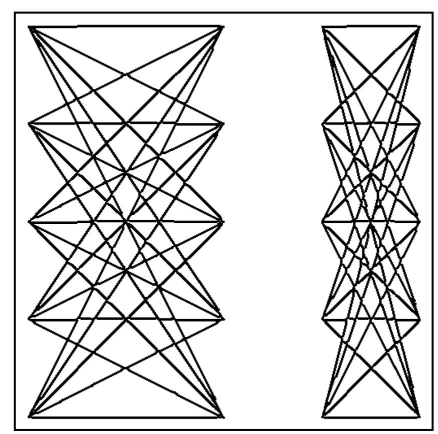
Symmetry



Imperfect Symmetry

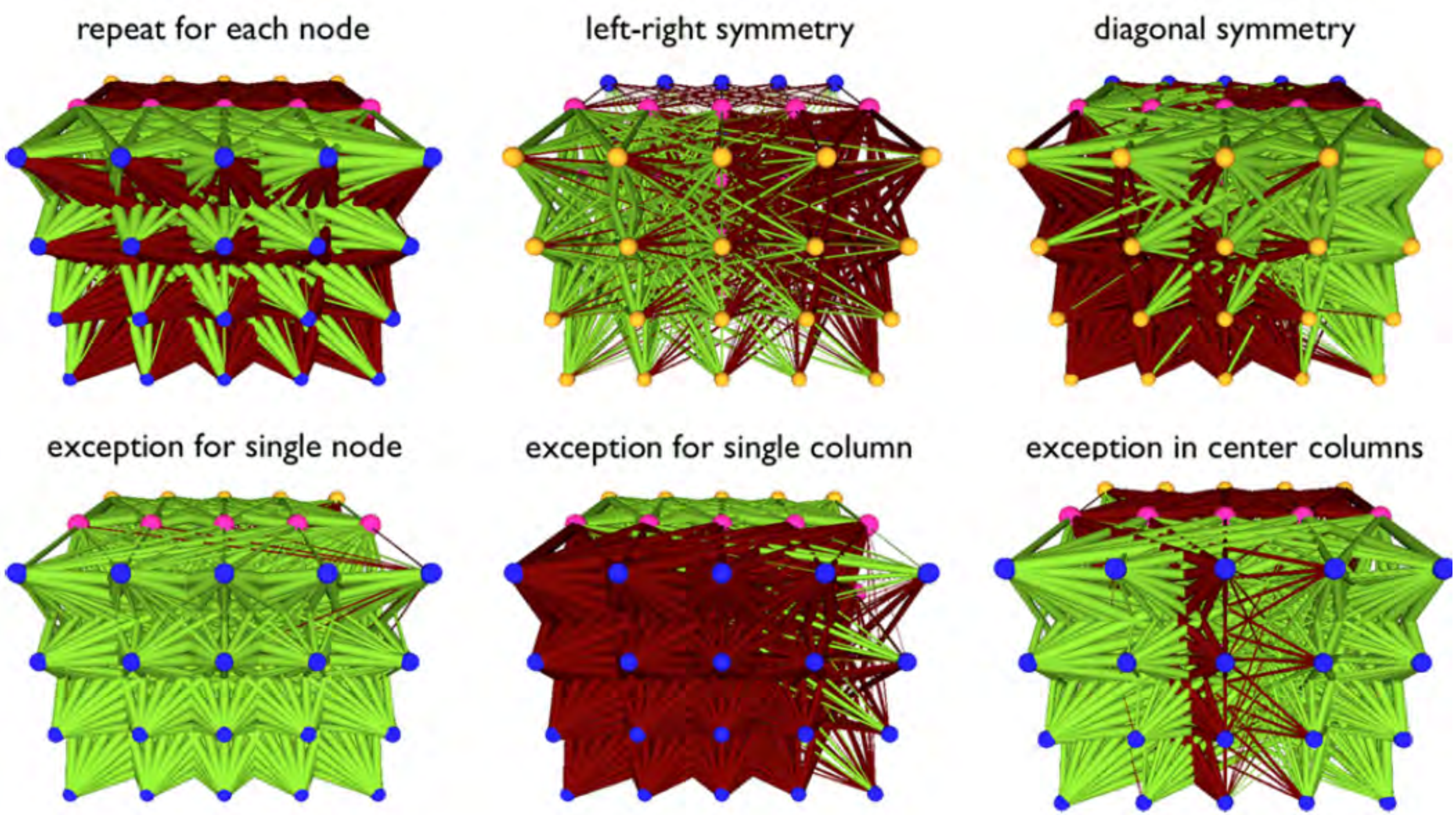


Repetition



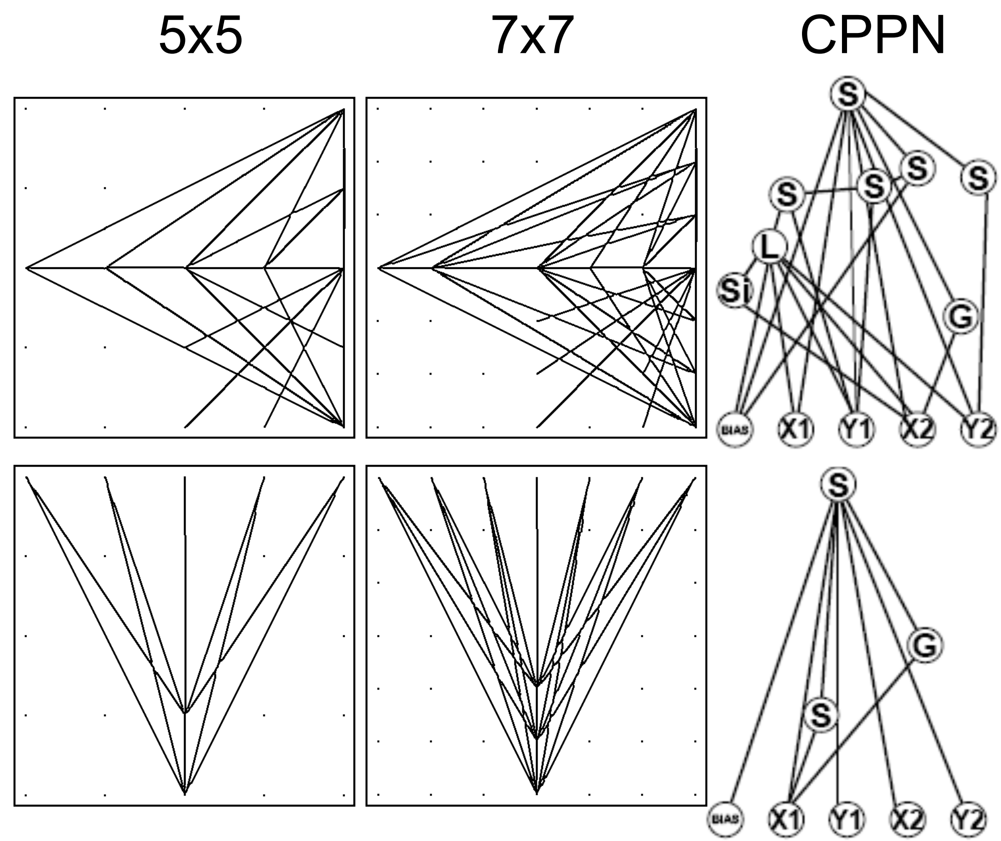
Repetition with Variation

Fundamental Regularities Produced by 6-D CPPNs



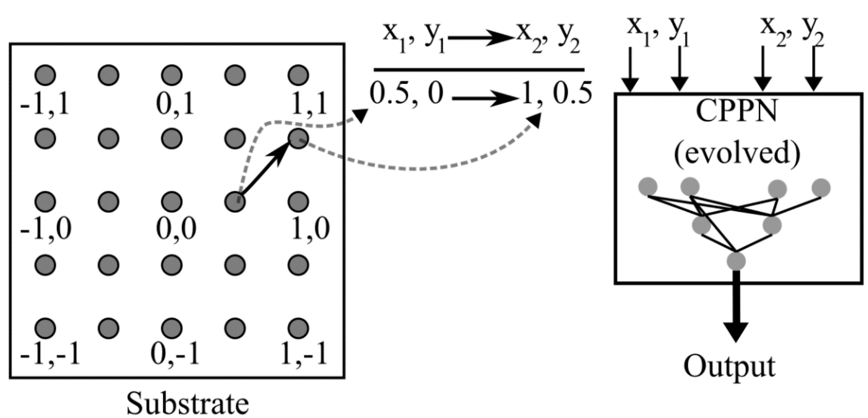
Resolution Independence

- CPPN learns a *connectivity concept*, **not** individual connections
- Concepts at 5x5 and 7x7 nodes
- Intuitive expansion of the pattern
- A novel capability
- NN can be scaled to higher resolutions

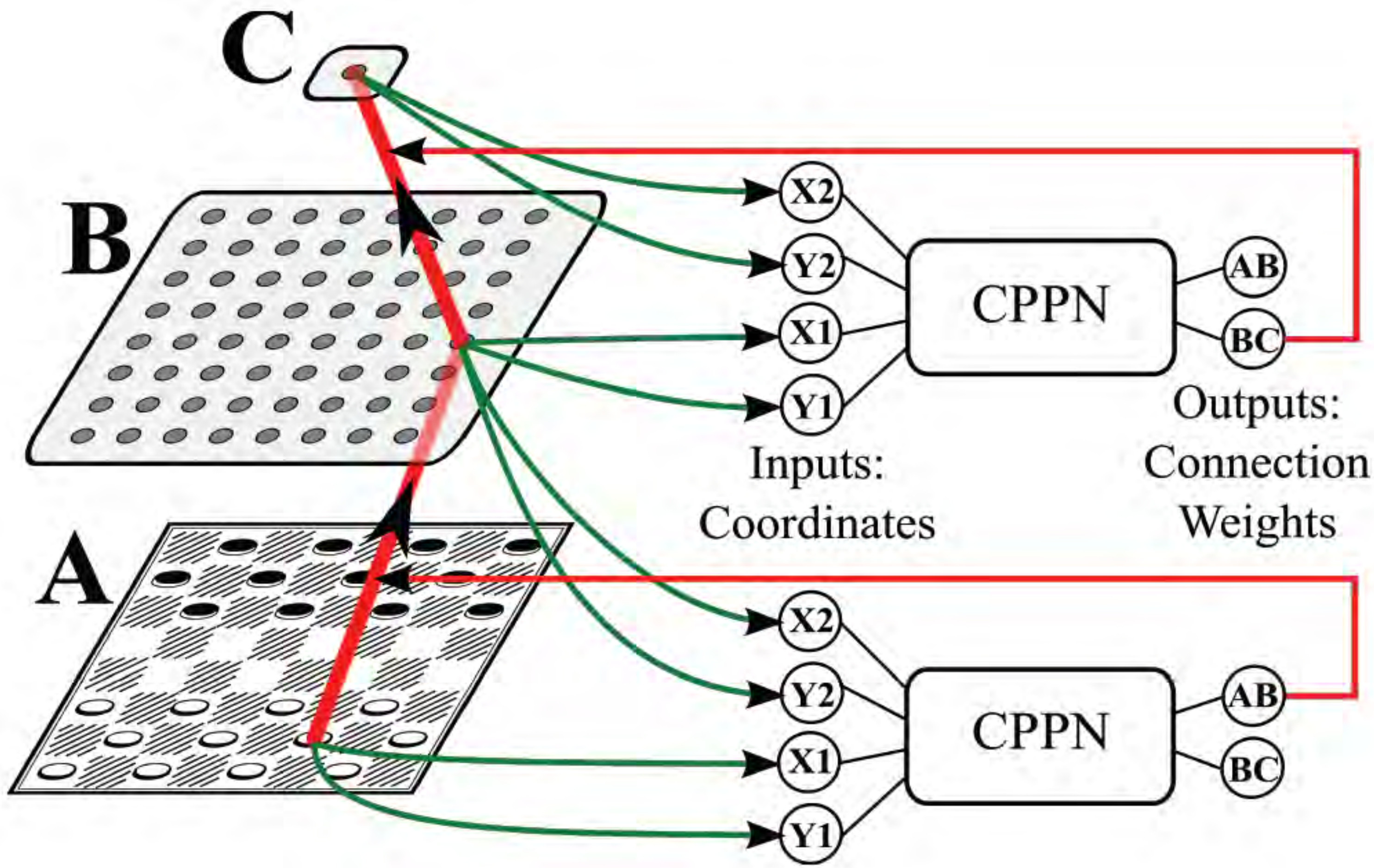


CPPNs “See” Geometry

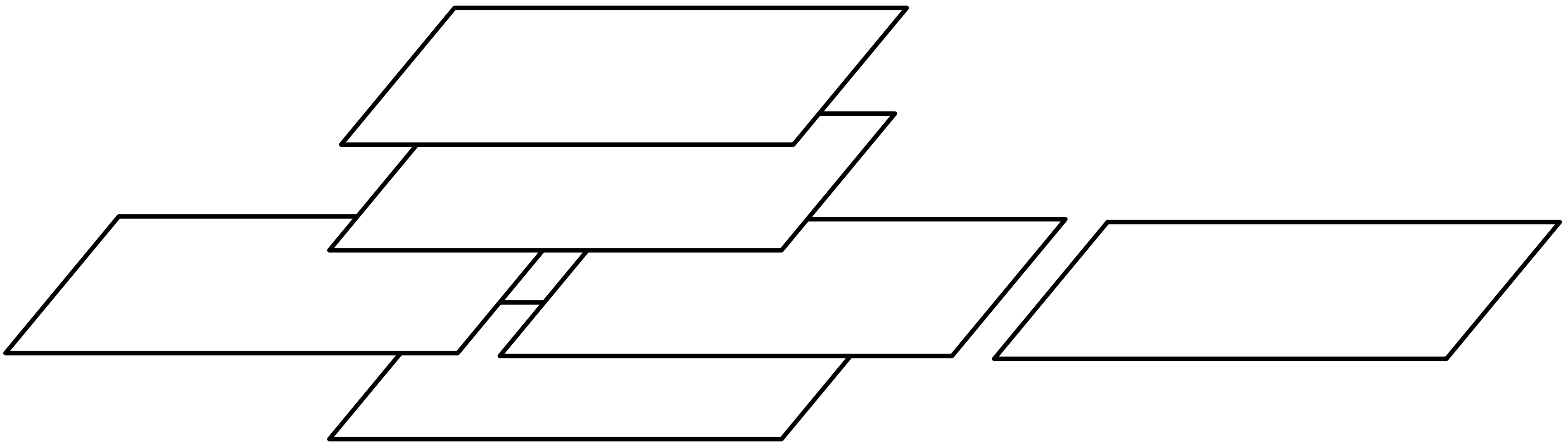
- The CPPN generates the network as a *function* of the substrate geometry
  - Instead of building in a mechanism for processing geometry (e.g. convolution)…
  - **Build a representation that can *discover the mechanism*!**



Multilayer Sandwich Geometry (e.g. in Checkers)

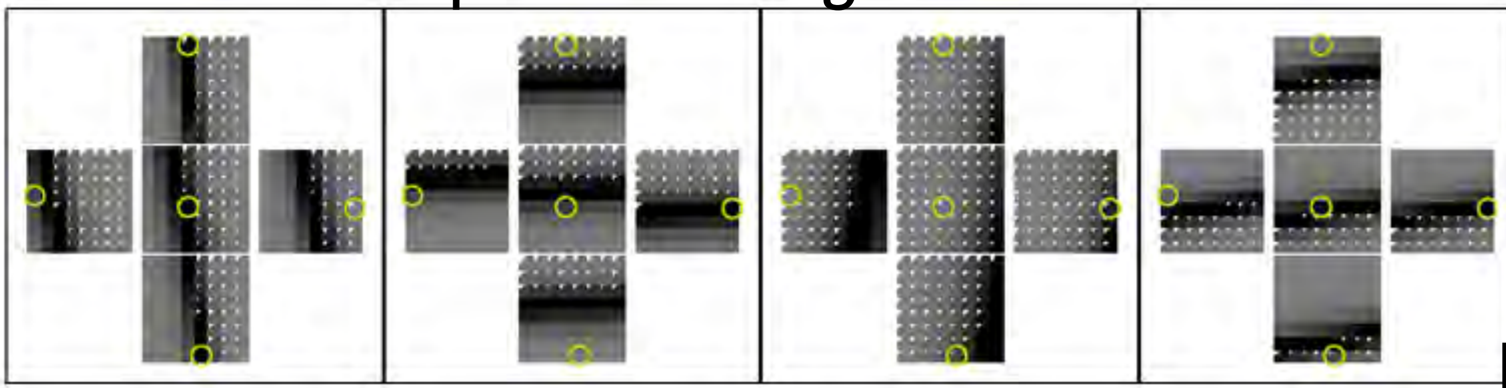


Can Contain Multiple “Filters”



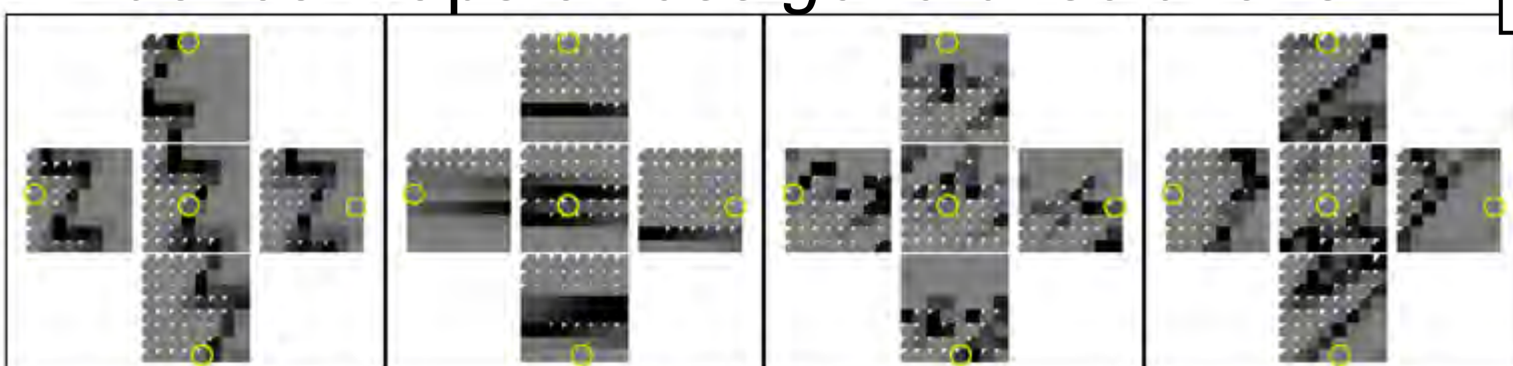
Geometric Patterns Inside HyperNEAT Checkers NNs

Influence Maps of *more general* solutions



We can see the difference

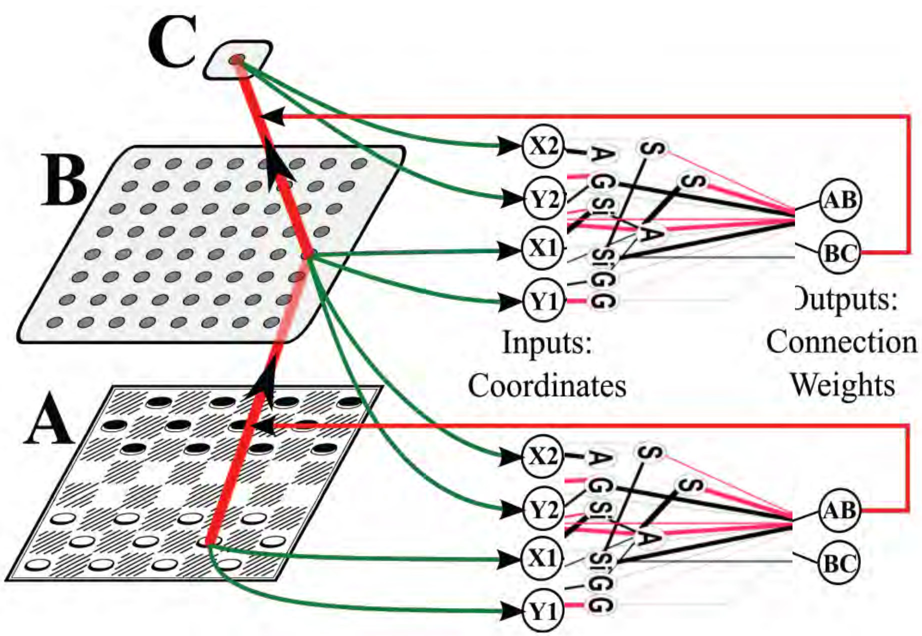
Influence Maps of *less general* solutions

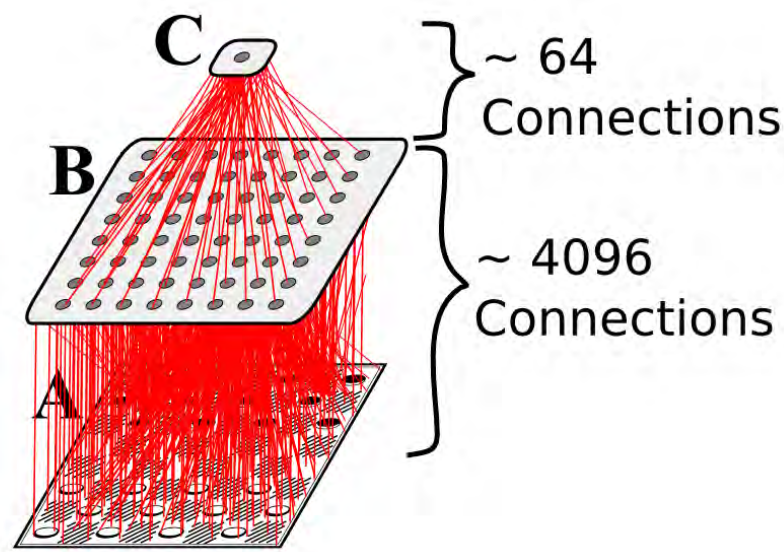


Jason Gauci and Kenneth O. Stanley (2010). Autonomous Evolution of Topographic Regularities in Artificial Neural Networks. In: Neural Computation journal 22(7), pages 1860-1898. Cambridge, MA: MIT Press.

Compression and Search

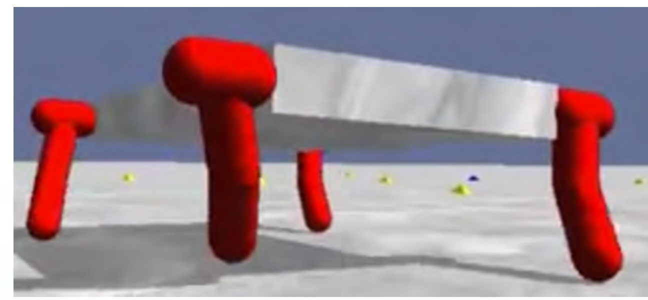
- Why indirect encoding can succeed quickly
  - Searches a compressed space (CPPNs)
    - Lower-dimensional

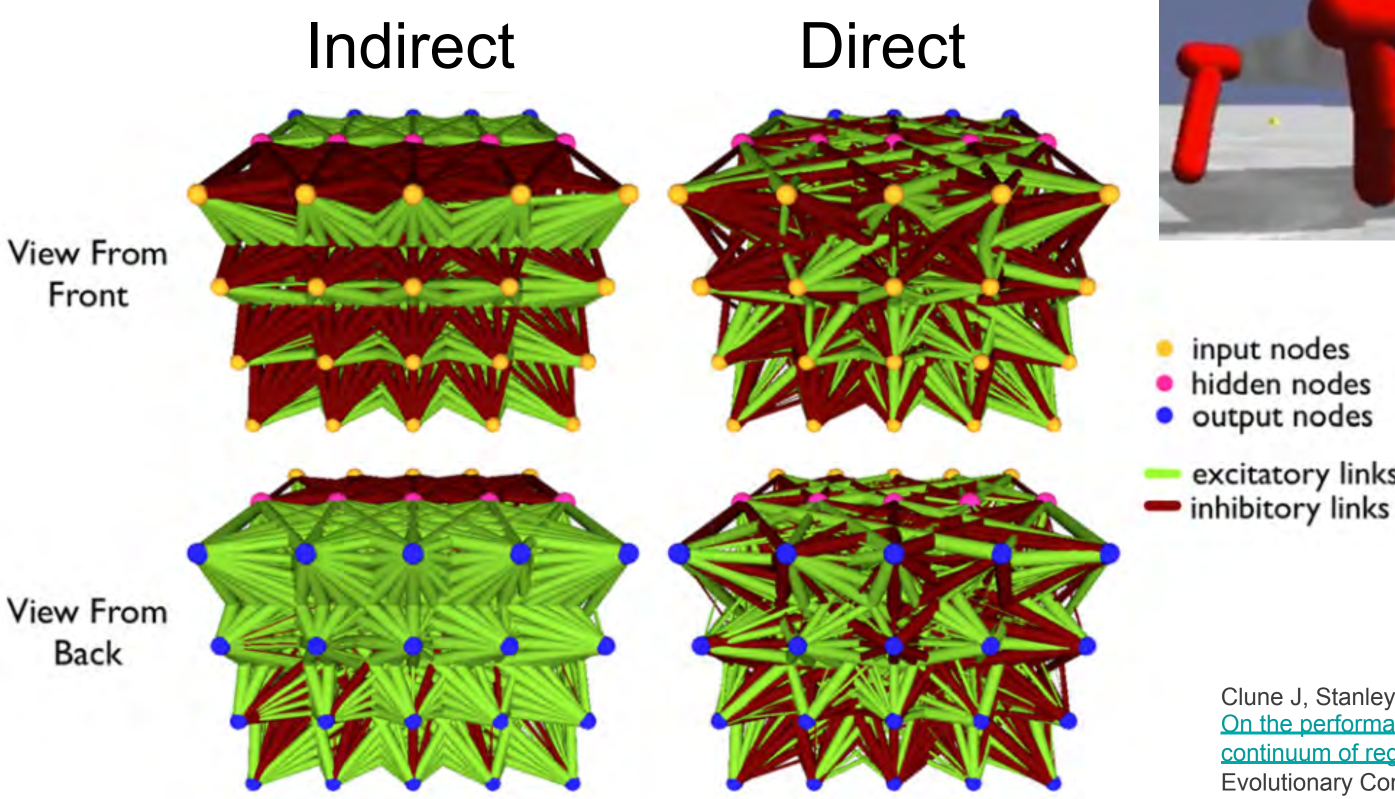




Regularity is Fundamental to Real World Problems

- Gait generation: far more effective through CPPN-generated networks





Clune J, Stanley KO, Pennock RT, Ofria C (2011) On the performance of indirect encoding across the continuum of regularity IEEE Transactions on Evolutionary Computation. 15(3): 346-367

CPPN-based NNs Are Differentiable

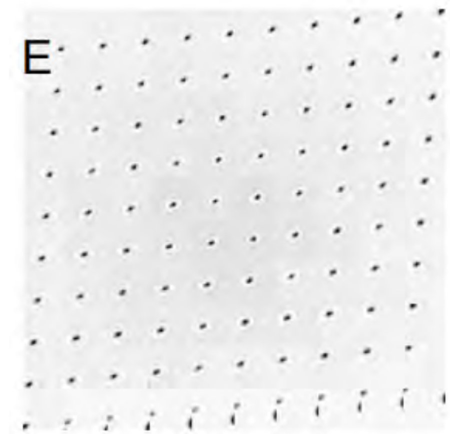
- Multiple realizations
  - DPPNs (differentiable pattern producing networks; Fernando et al. 2016)
  - Hypernetworks (Ha et al. 2016)
  - GENIE (geometrically expressive network for indirect encoding): coming soon with some surprises about convolution!
- Regularity in visual processing
  - e.g. convolution

Regularity is Fundamental to Real World Problems

- CPPNs/DPPNs *discovered* convolution (it was not built in)
- A simple concept:

$$w(x_1, y_1, x_2, y_2) \equiv \tilde{w}(x_2 - x_1, y_2 - y_1)$$

- But can indirect encoding discover *beyond* convolution? $w(x_2 - x_1, y_2 - y_1, x_1, y_1)$
  - E.g. repetition with variation
  - Like the "relaxed weight sharing" in LSTMs generated by hypernetworks

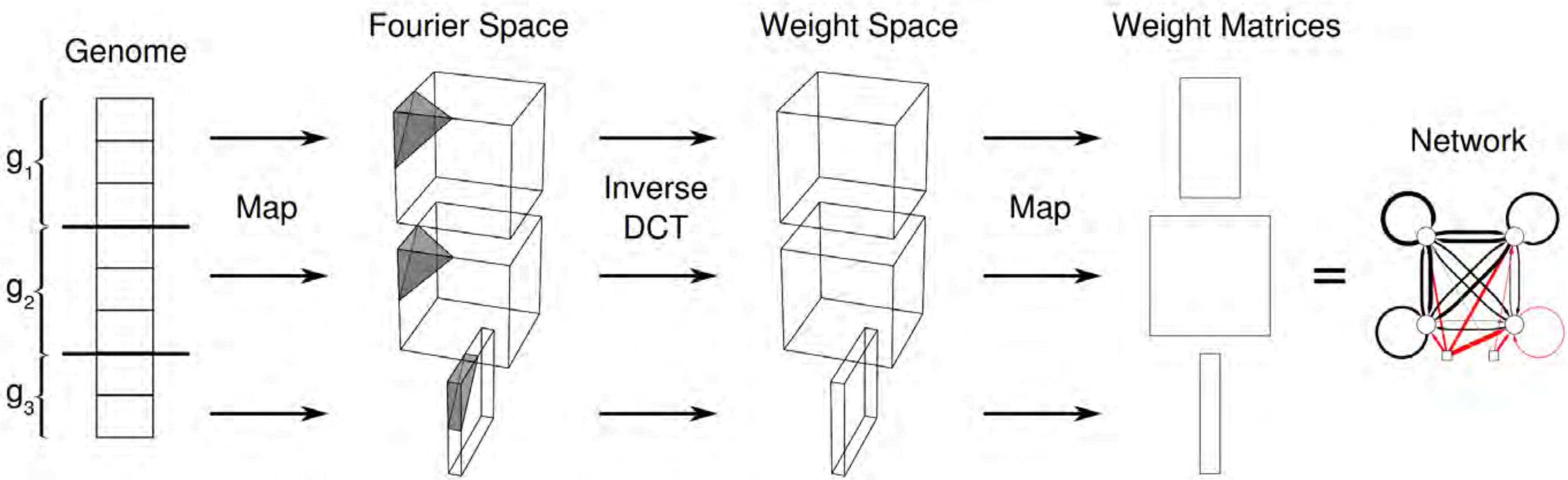


Fernando, Chrisantha, Dylan Banarse, Malcolm Reynolds, Frederic Besse, David Pfau, Max Jaderberg, Marc Lanctot and Daan Wierstra. "Convolution by Evolution: Differentiable Pattern Producing Networks." *GECCO* (2016).

Ha, David & Dai, Andrew & V Le, Quoc. (2017). HyperNetworks. ICLR (2017)

Alternative CPPN-like Encodings

Koutnik, Jan and Cuccu, Giuseppe and Schmidhuber, Juergen and Gomez, Faustino (2013) *Evolving Large-Scale Neural Networks for Vision-Based TORCS.* In: Foundations of Digital Games, 14-17/05/2013, Chania, Crete.



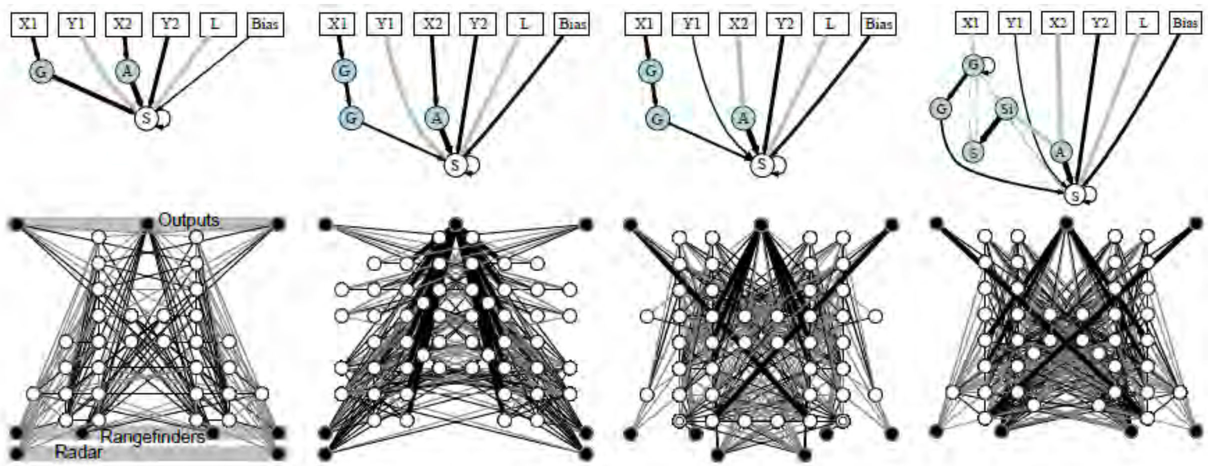
- Wavelet-based alternative representation to CPPNs from Koutnik et al. 2013
- Encodes million-conection NN that learns to drive

Interesting Extensions

- Architecture search: describe through CPPN
- Substrate evolution and architecture search: Automate everything
  - ES-HyperNEAT, “Deep HyperNEAT”

Felix A. Sosa and Kenneth O. Stanley (2018). *Deep HyperNEAT: Evolving the Size and Depth of the Substrate.* Evolutionary Complexity Research Group Undergraduate Research Report, University of Central Florida Department of Computer Science

Sebastian Risi and Kenneth O. Stanley (2012) *An Enhanced Hypercube-Based Encoding for Evolving the Placement, Density and Connectivity of Neurons.* Artificial Life journal. Cambridge, MA: MIT Press, 2012.



- Adaptation: CPPN as a universal learning rule
  - $CPPN(x_1,y_1,a_1,x_2,y_2,a_2)$ = delta_w: Universal learning rule!
  - Rules of adaptation *themselves* can be spread in a pattern

Risi, Sebastian, and Kenneth O. Stanley. "A unified approach to evolving plasticity and neural geometry." *The 2012 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2012.

Looking Forward

How will we achieve our most ambitious goals?

- Our ambitious goal: AGI
- How will we get there?
- Do the lessons from this tutorial help?

Manual Path to AI

- Dominant paradigm in ML
- Phase 1: Identify key building blocks

Key Building Blocks?

how many more?
hundreds? thousands?

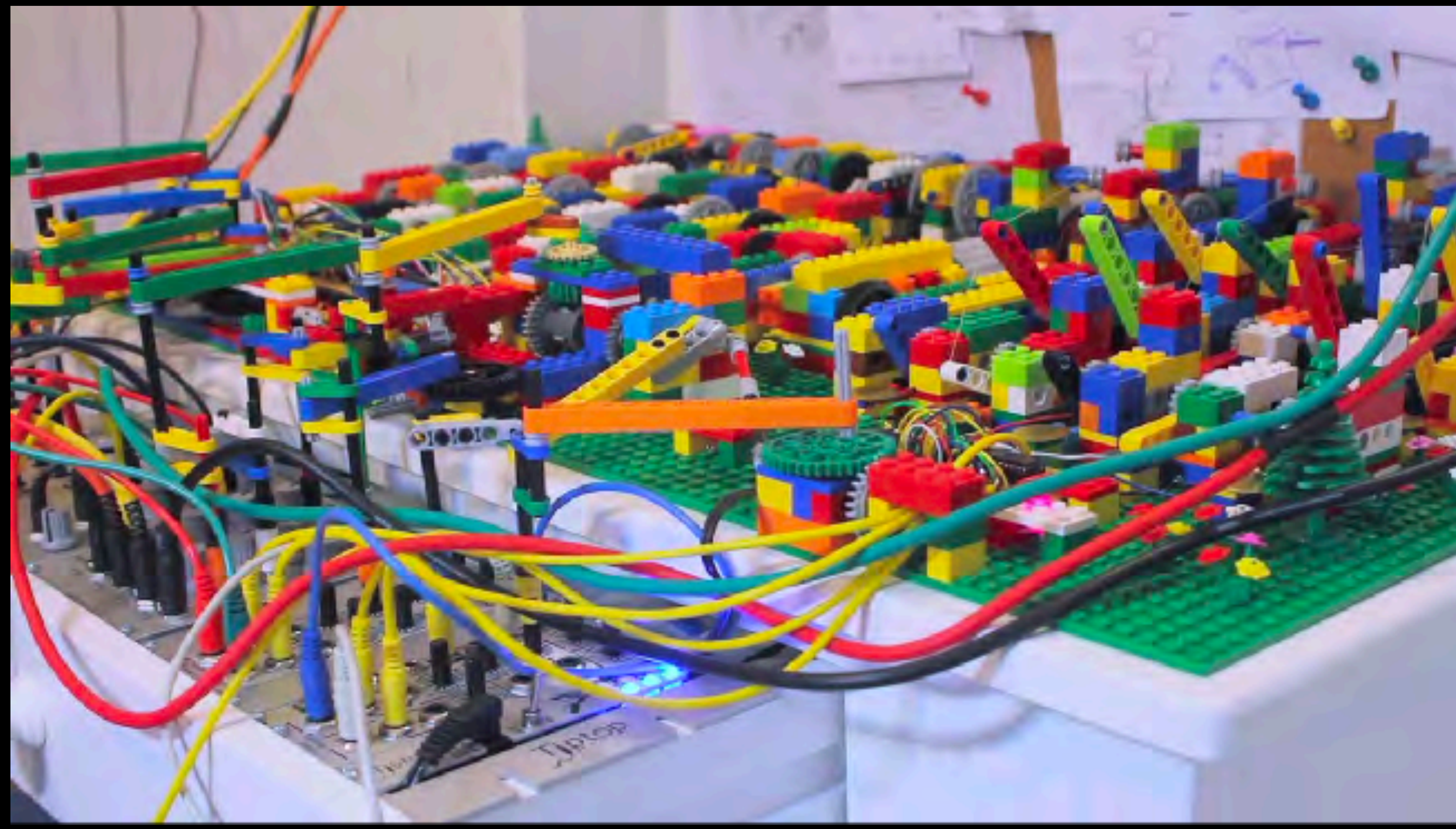
can we find them all?

- convolution
- attention mechanisms
- spatial tranformers
- batch/layer norm
- a learned loss (e.g. evolved policy gradients)
- hierarchical RL, options
- structural organization (regularity, modularity, hierarchy)
- intrinsic motivation (many different flavors)
- auxiliary tasks (predictions, autoencoding, predicting rewards, etc.)
- good initializations (Xavier, MAML, etc.)
- catastrophic forgetting solutions
- universal value functions
- hindsight experience replay
- LSTM cell machinery variants
- complex optimizers (Adam, RMSprop, etc.)
- Dyna
- variance reduction techniques
- activation functions
- good hyperparameters
- capsules
- gradient-friendly architectures (skip connections, highway networks)
- value functions, state-value functions, advantage functions
- recurrence (where?)
- multi-modal fusion
- models
- trust regions
- Bayesian everything
- Active learning
- Probabilistic models
- Distance metrics (latent codes)
- etc.

Manual Path to AI

- Dominant paradigm in ML
- Phase 1: Identify key building blocks

- Phase 2: Combine building blocks into complex thinking machine
  - Herculean task
  - Is it possible?



Overall Machine Learning Trend: Learn the Solution

- Features
  - HOG/SIFT → Deep Learning
- Architectures
  - Hand designed → Learned
- Hyperparameters & data augmentation
  - Manually tuned → Learned
- RL algorithms
  - Hand designed → Meta-learning

suggests alternate path

AI-Generating Algorithms

Clune 2019

- Learn as much as possible
- Bootstrap from simple to AGI
- Expensive outer loop
  - produces a sample-efficient, intelligent agent for inner loop
- We know it works
  - occurred on Earth

AI-Generating Algorithms

Clune 2019

Three Pillars

1. Meta-learn architectures
2. Meta-learn learning algorithms
3. Generate effective learning environments

AI-Generating Algorithms

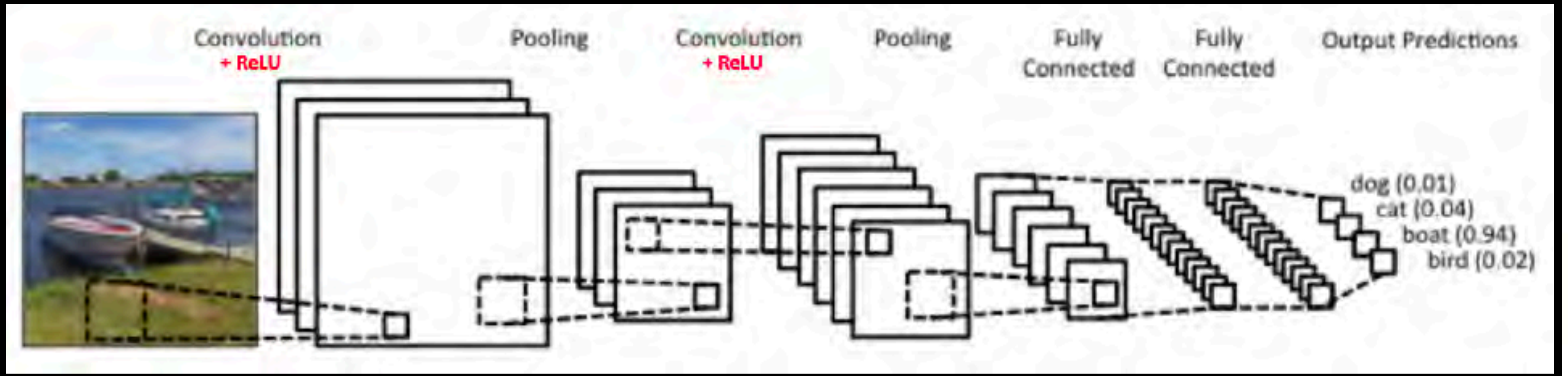
Clune 2019

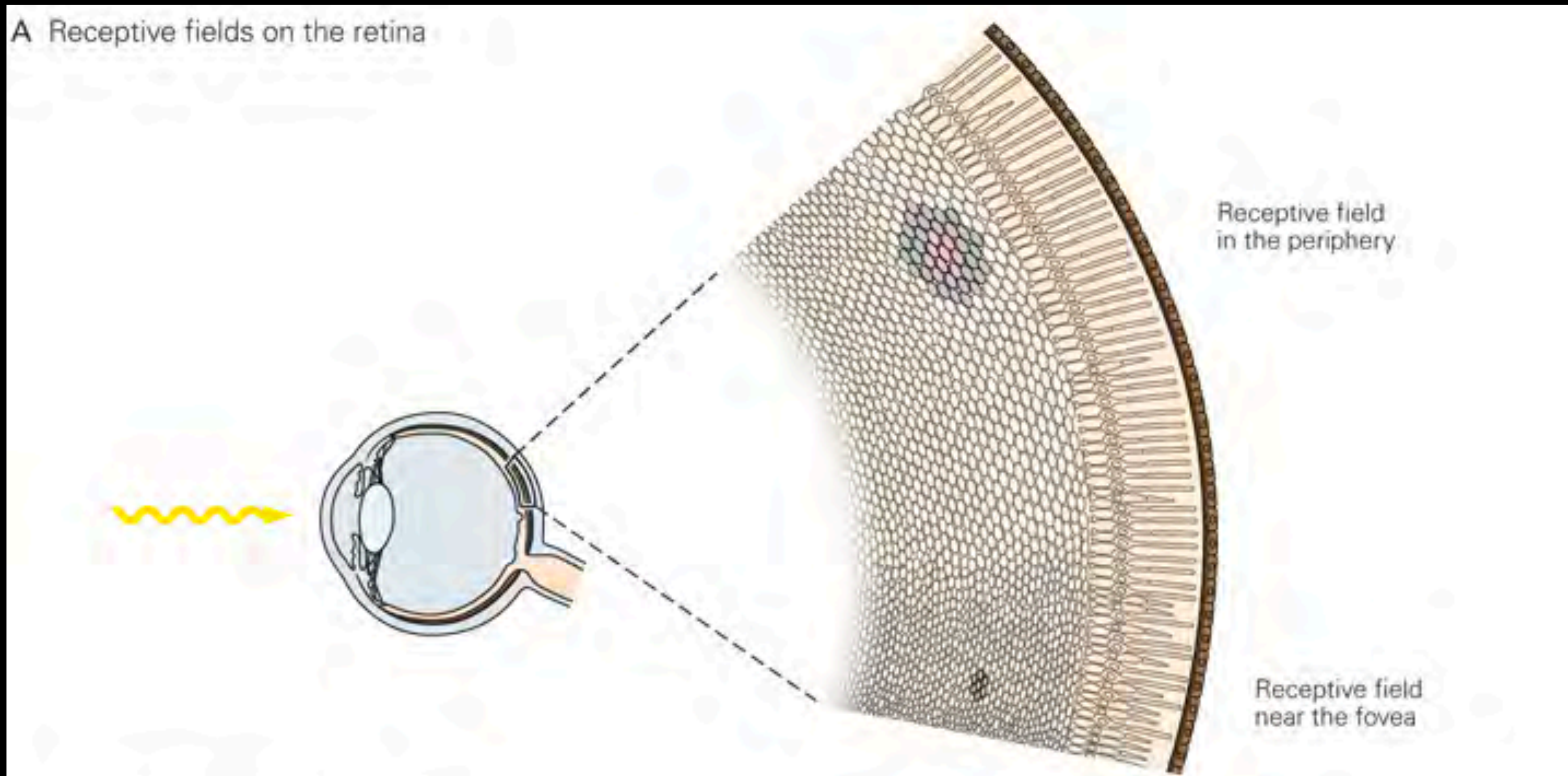
- Three Pillars
  1. Meta-learn architectures
  2. Meta-learn learning algorithms
  3. Generate effective learning environments

Indirect Encoding

Open-Ended Search

Quality Diversity





AI-Generating Algorithms

Clune 2019

- Three Pillars
  1. Meta-learn architectures
  2. Meta-learn learning algorithms
  3. Generate effective learning environments

Indirect Encoding

Open-Ended Search

Quality Diversity

AI-Generating Algorithms

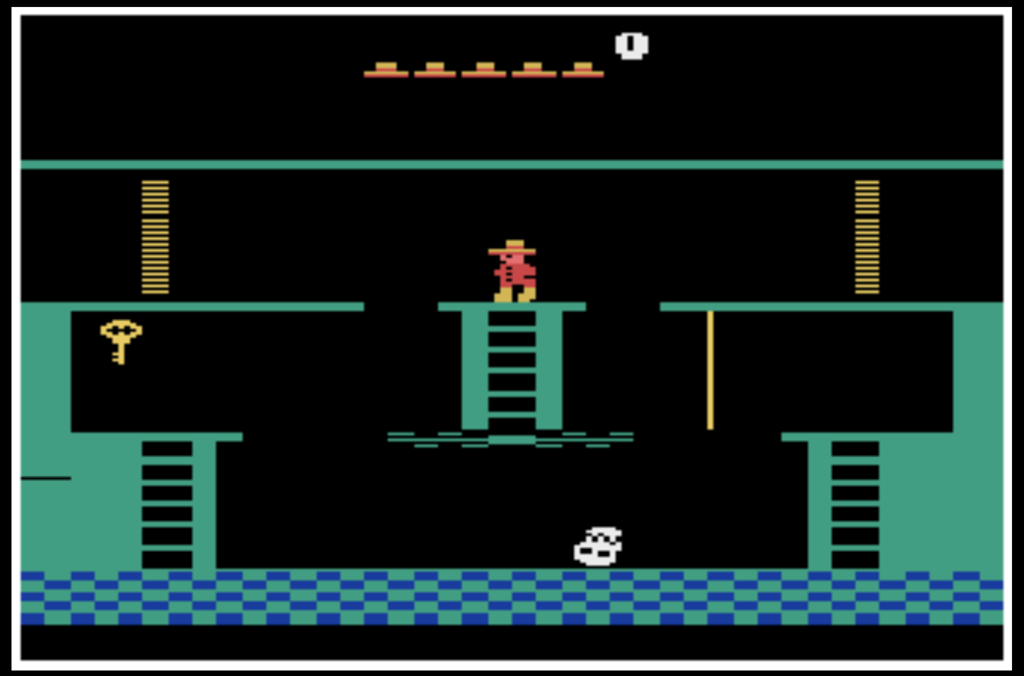
Clune 2019

- May be fastest path to AGI
- Interesting even if not
  - how simple processes to bootstrap into intelligence
    - necessary, sufficient, catalyzing factors
  - understand our origins
  - likelihood of such processes occurring elsewhere in the universe
- Grand challenge of CS

Conclusions

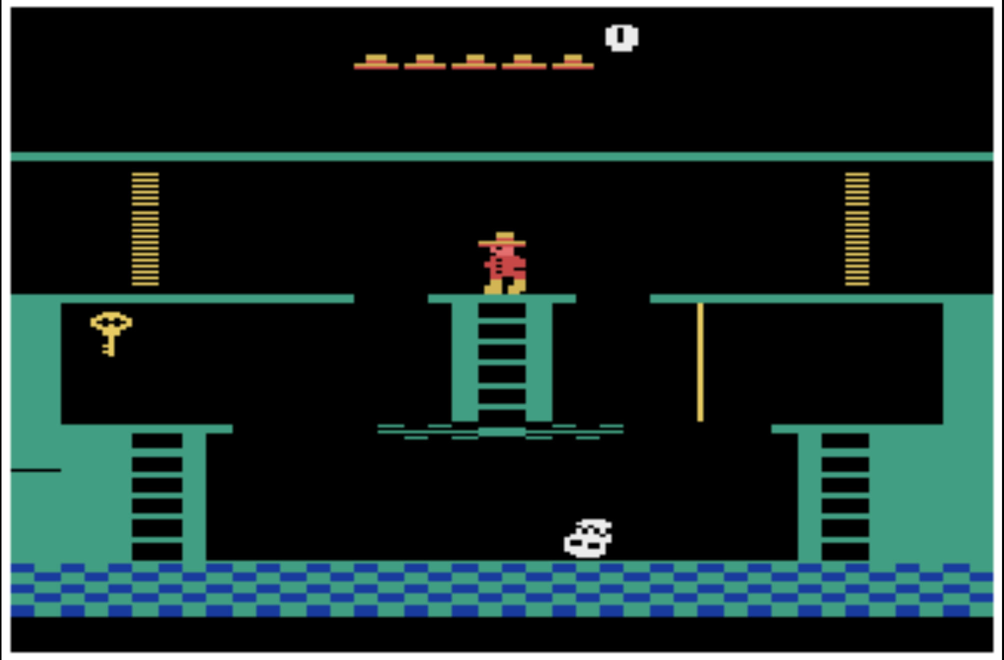
Conclusions

- Novelty Search
- Quality Diversity
- Open-Ended Search
- Indirect Encoding



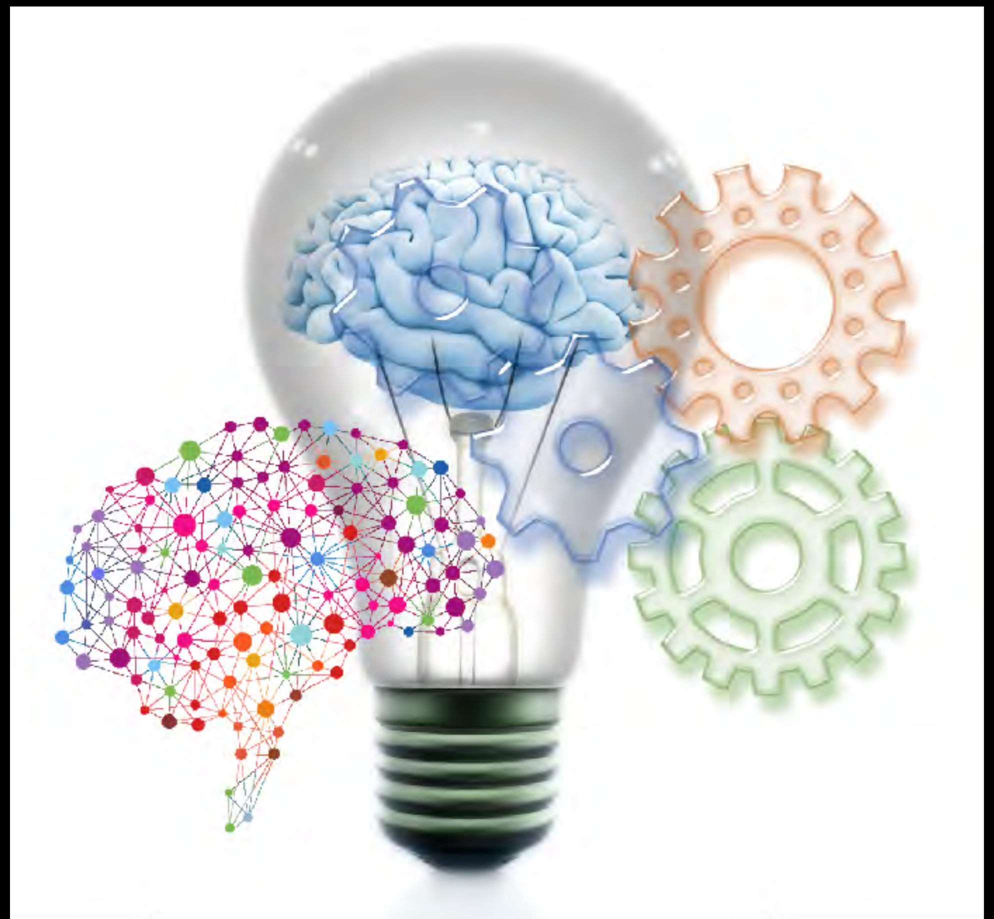
Conclusions

- interesting, powerful ideas
  - help solve previously unsolvable problems
  - introduce entirely new types of problems
- Grand challenges
  - Open-ended algorithms
  - AI-generating algorithms



Conclusions

- Whether descendant or convergent, lots of these ideas are being hybridized with machine learning to great effect
  - HER, DIAYN, Go-Explore, PBT/AlphaStar, HyperNetworks, etc.
- Potential for lots more!
  - How might these ideas help with your techniques?
- Might help us achieve our most ambitious research goals



Recommended Reading

PDFs available on our websites

- Stanley KO, Clune J, Lehman J, Miikkulainen R (2019) Designing Neural Networks through Neuroevolution. Nature Machine Intelligence, 1:1, 24-35.
  - Reviews most of the concepts in the tutorial and provides cites to the original papers, including: Novelty Search, Novelty Search with Local Competition, MAP-Elites, Intelligent Intelligent Trial & Error, Evolutionary Strategies + Novelty Search, Quality Diversity, Innovation Engines, CMOEA, NEAT, CPPNs, HyperNEAT, Indirect Encoding, Minimal criterion coevolution
- Open-endedness: The last grand challenge you've never heard of. Stanley, Lehman, Soros. 2017. https://www.oreilly.com/ideas/open-endedness-the-last-grand-challenge-youve-never-heard-of
- AI-GAs: AI-generating algorithms, an alternate paradigm for producing general artificial intelligence. (2019) Clune. https://arxiv.org/abs/1905.10985
- Ecoffet A, Huizinga J, Lehman J, Stanley KO, Clune J (2019) Go-Explore: a New Approach for Hard-Exploration Problems. arXiv 1901.10995.
- Wang R, Lehman J, Clune J, Stanley KO (2019) Paired Open-Ended Trailblazer (POET): Endlessly Generating Increasingly Complex and Diverse Learning Environments and Their Solutions. arXiv 1901.01753.
- Autonomous skill discovery with Quality-Diversity and Unsupervised Descriptors. Cully 2019. arXiv:1905.11874, 2019
- Why Greatness Cannot Be Planned. Stanley & Lehman. 2015.