

Novelty Search in Evolutionary Computations

Arshiya Khan

CCRG

UD

Abandoning Objectives: Evolution through the Search for Novelty Alone

Joel Lehman and Kenneth O. Stanley
Evolutionary Complexity Research Group
School of Electrical Engineering and Computer Science
University of Central Florida
Orlando, FL 32816 USA
{jlehman,kstanley}@eecs.ucf.edu

In:

Evolutionary Computation journal, (19):2, pages 189-223, Cambridge, MA: MIT Press, 2011

Evolutionary computation

- Computational intelligence technique inspired from natural evolution
- Process of natural evolution is used as a role model for a strategy for finding optimal or near optimal solutions for a given problem
- Examples: Ant colony optimization, Swarm intelligence
- Genetic algorithms

Genetic algorithm

- High-quality solutions to optimization and search problems by relying on biologically inspired operators
- **Initial Population:** a genetic representation of the solution domain
- **Fitness Function:** Determines how fit an individual is (the ability of an individual to compete with other individuals)
- **Selection:** A pair of two individuals (parents) are selected based on their fitness scores
- **Crossover:** Offspring are created by exchanging the genes of parents among themselves until the crossover point is reached.
- **Mutation:** In certain new offspring formed, some of their genes can be subjected to a mutation with a low random probability.
- **Termination:** The algorithm terminates if the population has converged (does not produce offspring which are significantly different from the previous generation)

Population

bahama
abcdef
ijklmn
.....
mnopqr
stuvwx
cabana

Fitness objective

banana

Fitness score

banyan, 5
abcdef, 2 #(2 characters are
present in this word)
ijklmn, 1
mnopqr, 1
stuvwx, 0
cabana, 5

Selecting the Parents

banyan, 5
cabana, 5

Making a Crossover

canyan
cabyan
cabyna
babyna
banana
yanbac

All other possible
combinations from the parent
words 'banyan' and 'cabana'

Making a Mutation

banyan → yanyan,4
yanbac
yanbac → canbay,4
banyan → nanyab,5
abbyn → aabann,6
cabayn

Convergence

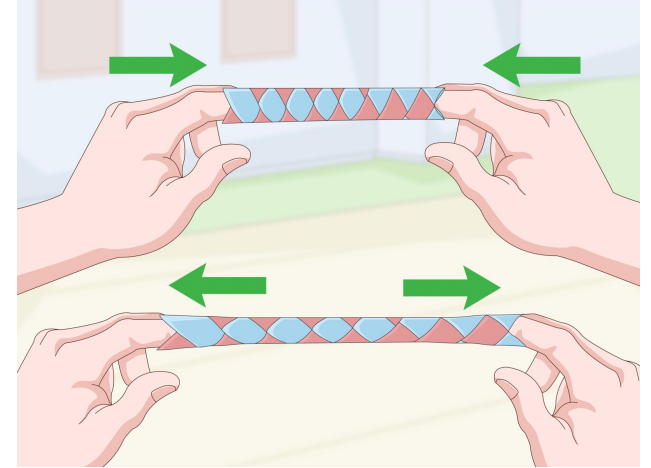
nnbaaa, 6
aaabnn, 6
aabann, 6
abaann, 6
banana, 6
baaann, 6

Objective function

- The fitness function normally measures progress towards an *objective* in the search space
- Similarity to ANN:
 - Reward moving closer to goal
 - Pathology of local optima (dead end)
- Increasing fitness does not always reveal the best path through the search space

Deception

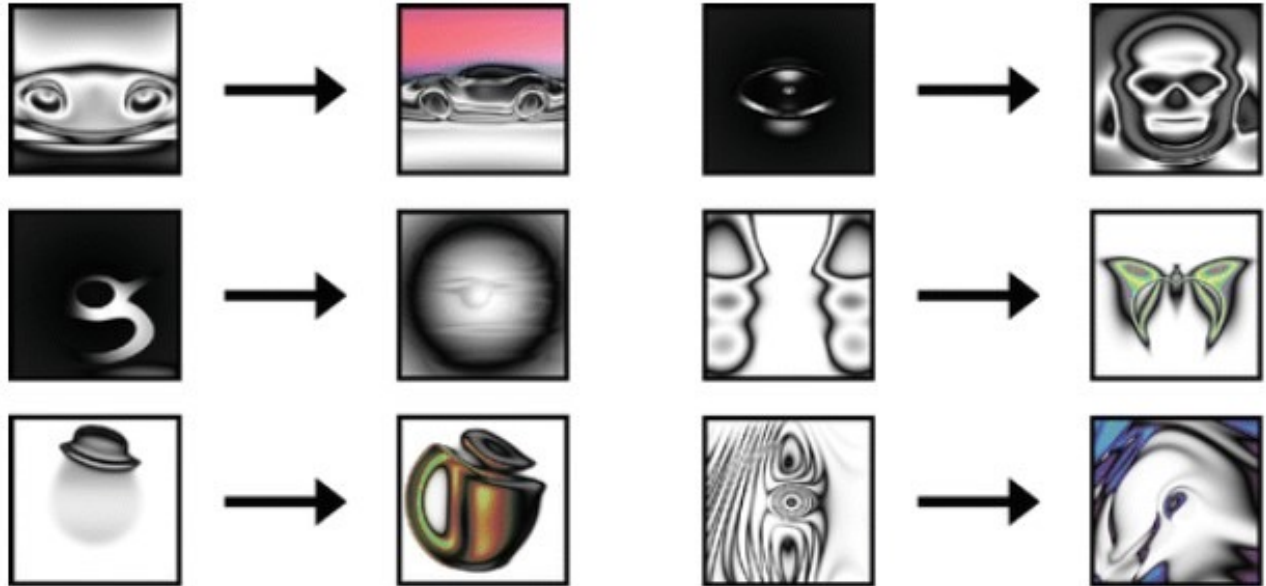
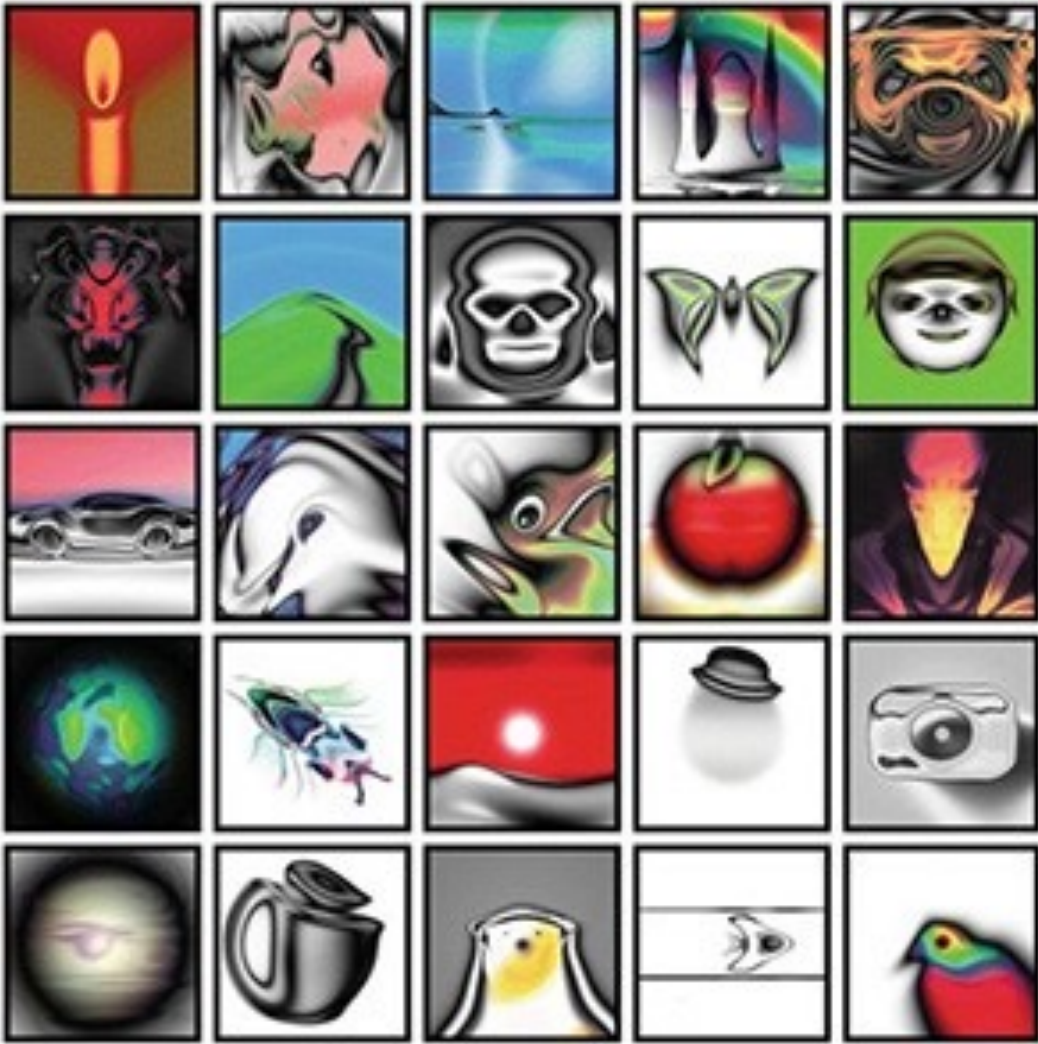
- **Chinese finger trap problem**
- Goal: Free one's fingers
- Rewarding solution: Pulling them apart
- Actual solution: Push one's fingers *together*, which seems to entrap them more severely
- A problem is deceptive if lower-order building blocks, when combined, do not lead to a global optimum



Non-objective Search

- Some objectives may be unreachable by direct objective-based search alone
- Systems without explicit objectives
- Mimic the unbounded innovation of natural evolution
- Open ended evolution
- **Novelty Search:** continually finding novel behaviors in the search space

<http://picbreeder.org/>

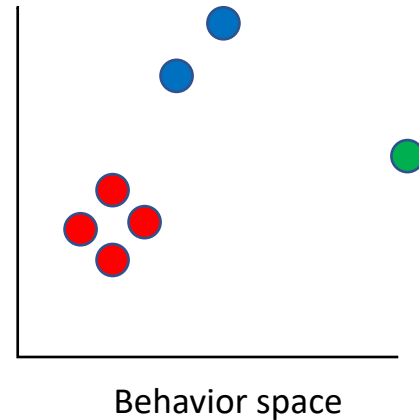


NEAT

- NeuroEvolution of Augmenting Topologies
- Objective based algorithm
- Evolve ANNs to solve difficult control and sequential decision tasks
- Evolution with a population of small, simple networks and *complexifies* the network topology into diverse species over generations

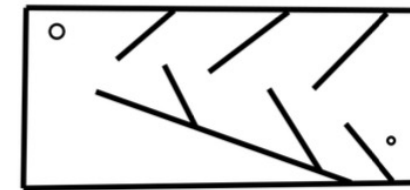
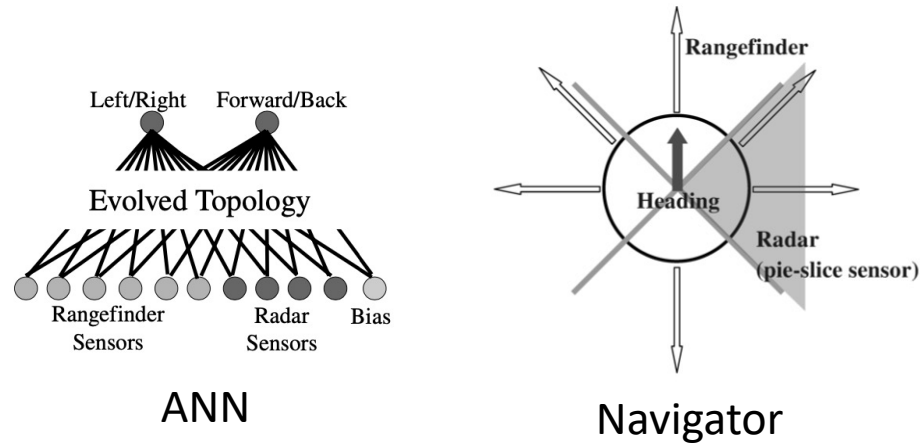
NEAT using novelty search

- Learning method is rewarded for finding any instance whose functionality is significantly different from what has been discovered before
- Novelty metric: In a space of unique behaviors, how far the new individual is from its predecessor. Example: k-nearest neighbor

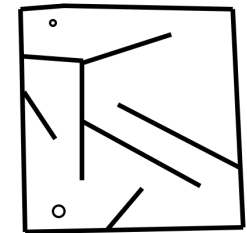


Maze experiment

- Suitable deceptive landscape
- Actor: A maze navigating robot controlled by an ANN
- Fitness function: How close the maze navigator is to the end point
- Deception: dead ends that lead close to the end point
- Goal: Searching for an *ANN* that itself can navigate the maze



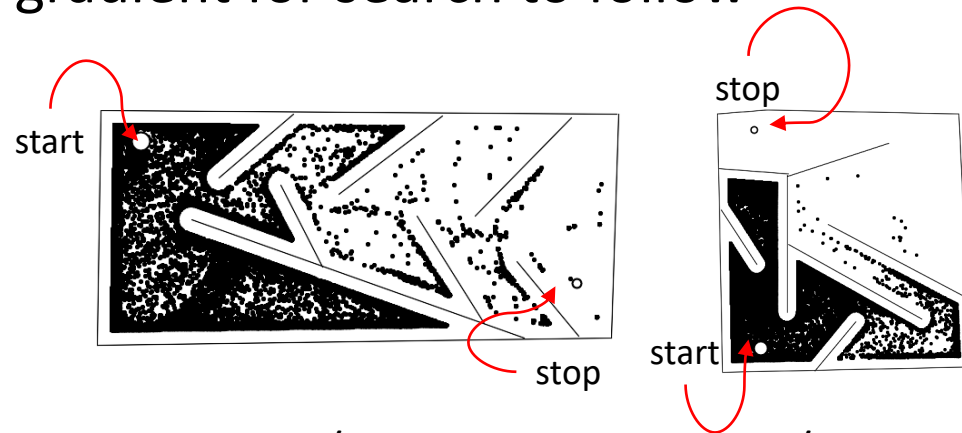
Medium map



Hard map

Fitness-based

- Fitness function: distance from the robot to the goal
- With no deceptive obstacles, this fitness function defines a monotonic gradient for search to follow



No of runs	40/40
No of steps	56, 334

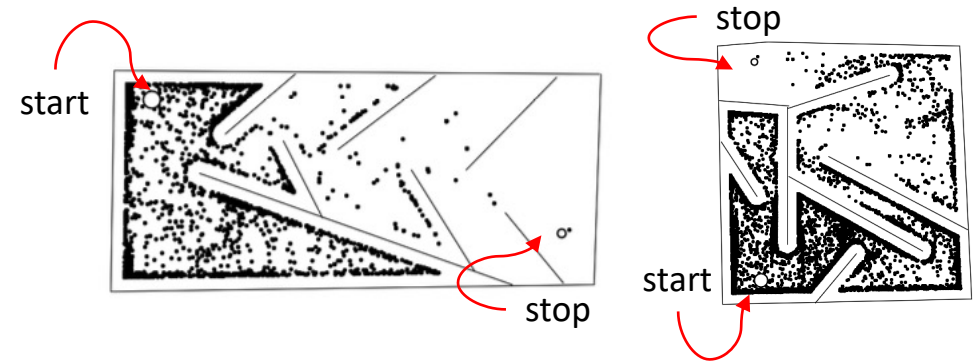
3/40

-

Dots = navigator end locations

Novelty-based

- Novelty metric rewards the robot for ending in a place where none have ended before
- Method of traversal is ignored
- No knowledge of the final goal



40/40

18, 274

39/40

35, 109

Questions??