Evolutionary algorithms for Controllers in Games

23.05.2019

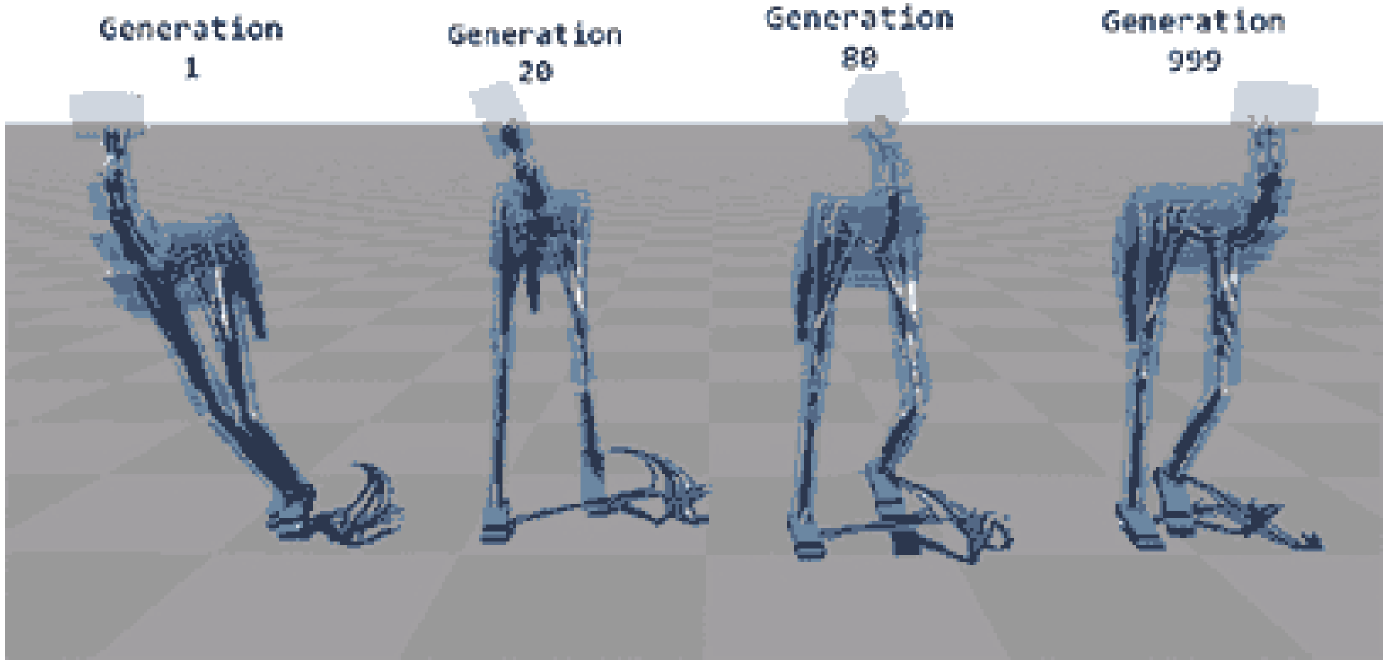
AI for Games

Oliver Mautschke

Overview

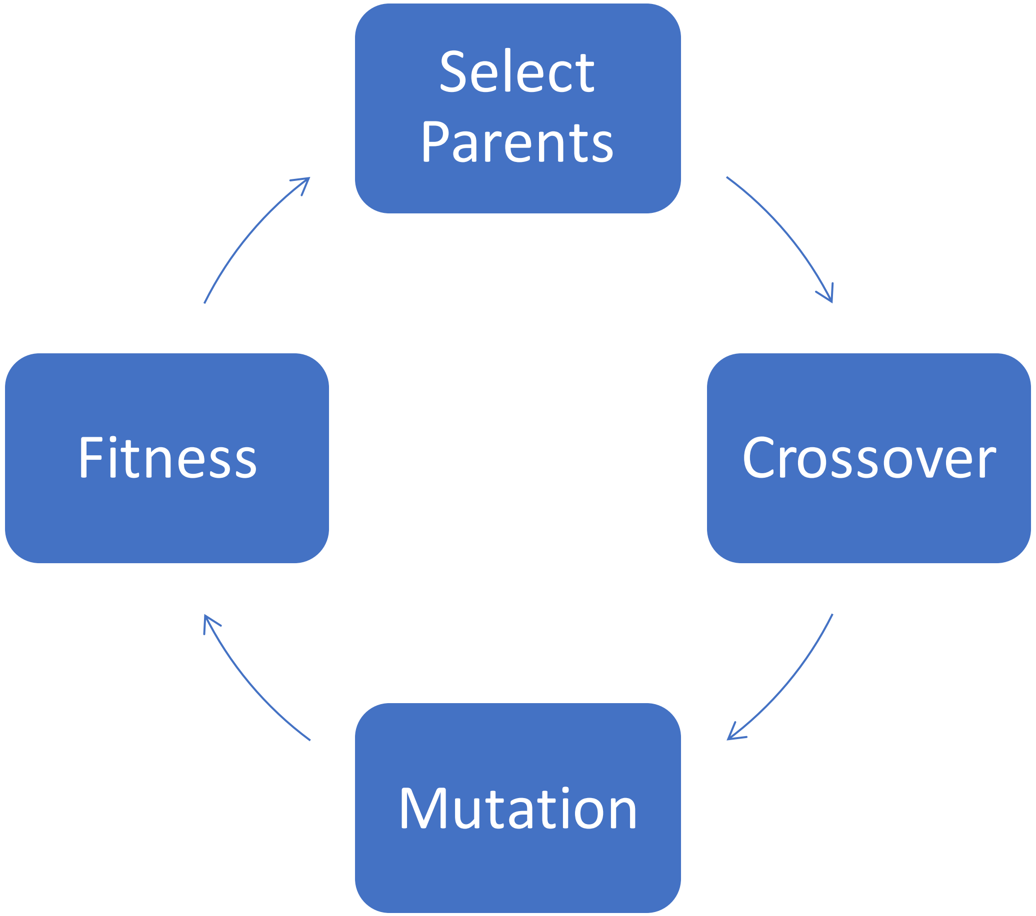
- Motivation
- Introduction to Evolutionary Algorithms
- Neuroevolution
- Evolving Behavior Trees
- Super Mario
- Conclusion

Motivation



An example for an use case of evolutionary algorithms [1]

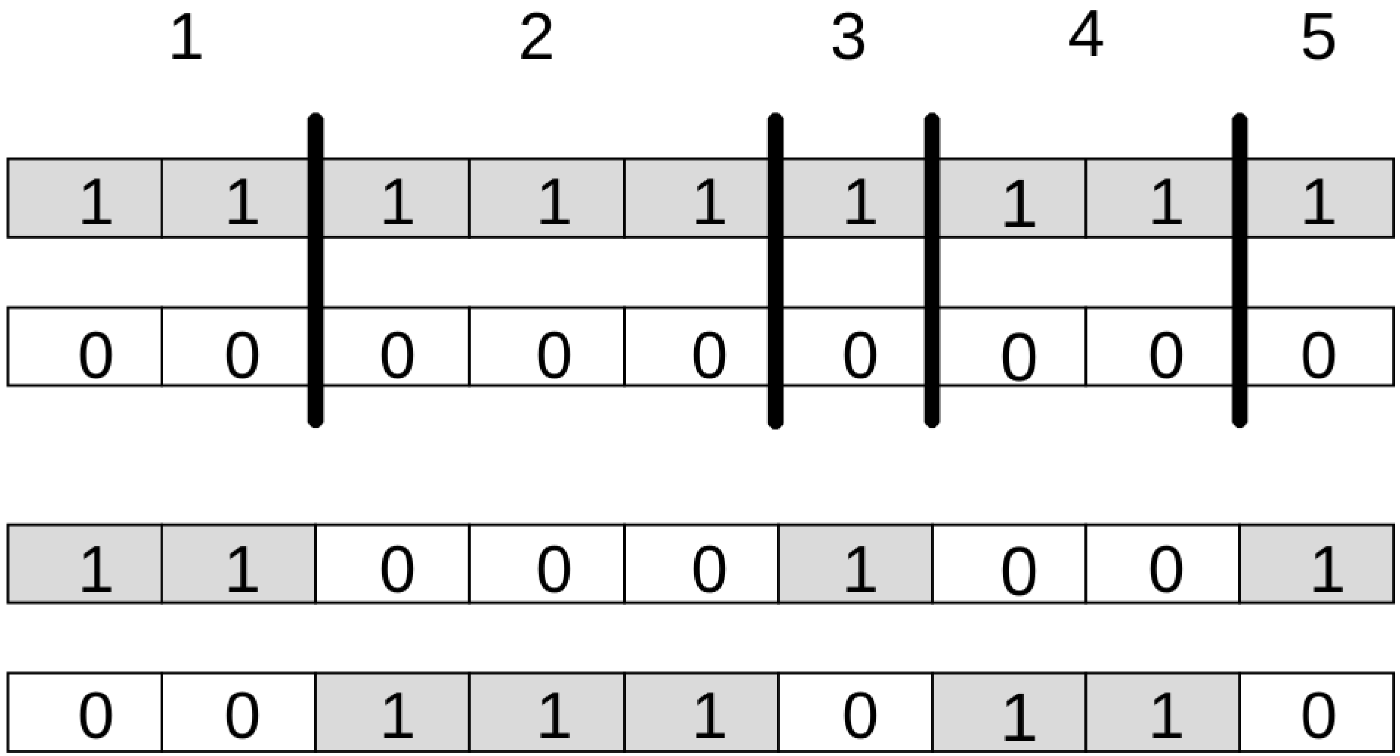
Introduction to Evolutionary Algorithms



Introduction to Evolutionary Algorithms

- Evaluate Fitness
  - Examples: Traveled Distance, Survived Time, Highscore
- Select Parents
  - Fortune Wheel, Tournament Selection
- Recombination
  - N-Point Crossover, Unified Selection
- Mutation
  - Bit-Flipping, Adding a delta

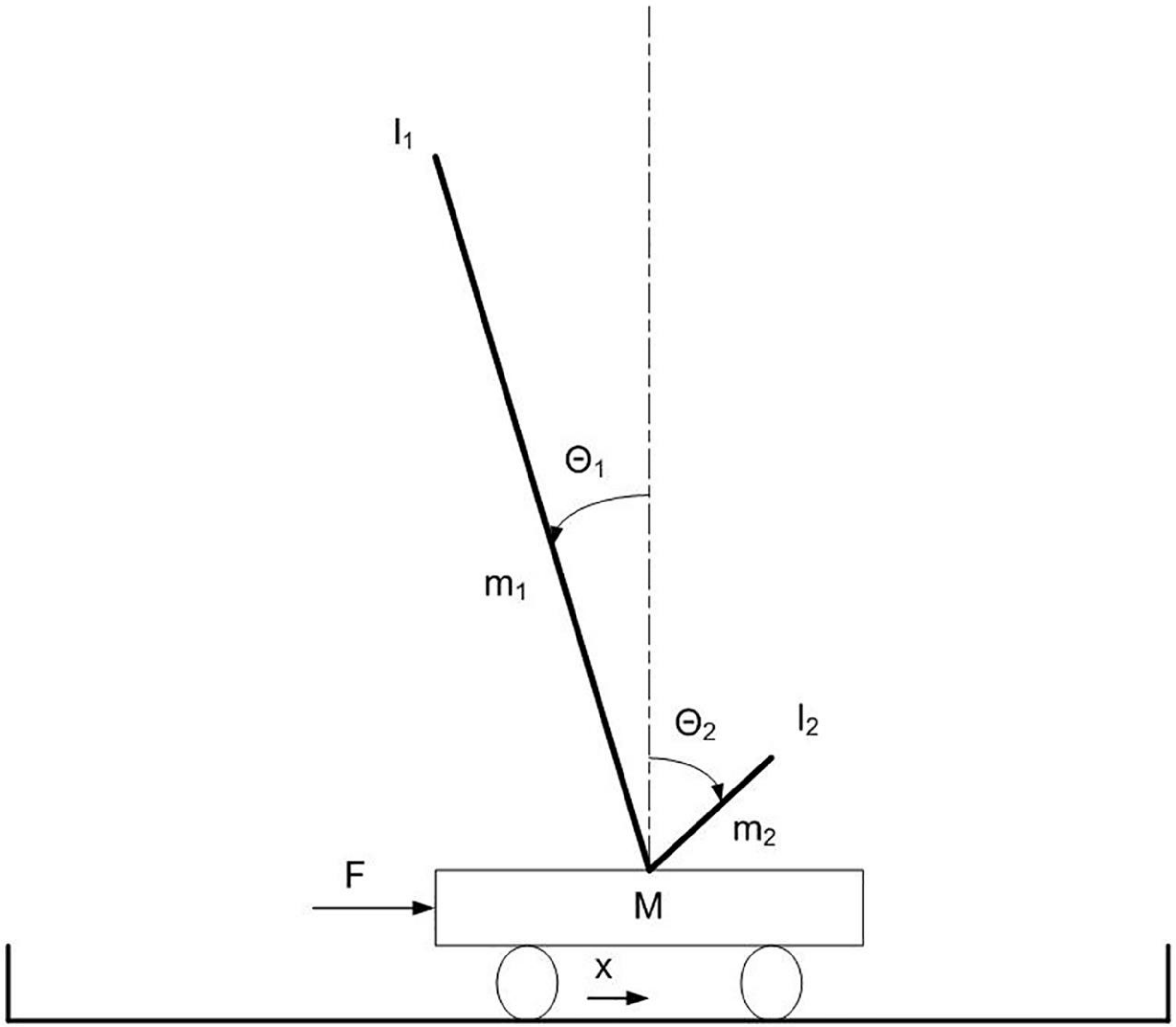
N-Point Crossover



N-Point Crossover [2]

Neuroevolution

- Double Pole Problem was THE Benchmark for Controller Problems
- There is no loss
- Archived Time is the fitness



Double Pole Balancing Problem [3]

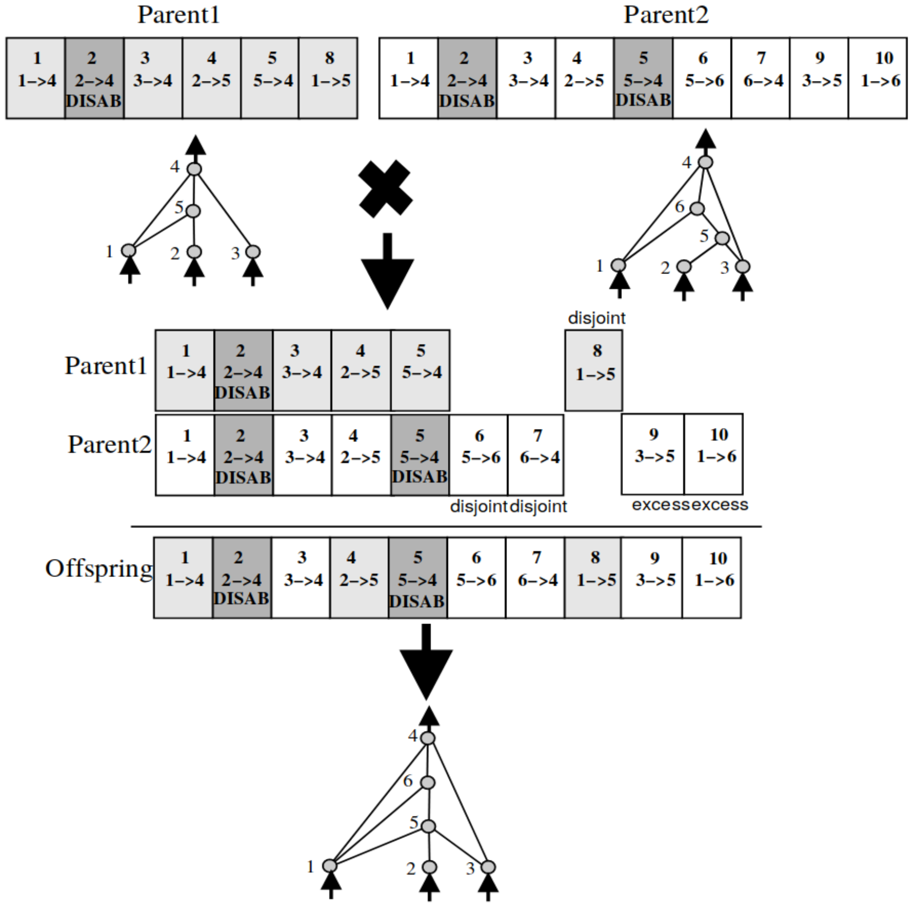
Neuroevolution – The Concept

- Encoding an ANN as a genome
- Applying genome to a task and measure their performance
  - The difference to "classical" optimization approaches for ANNs: Not the output loss is used, but the overall performance on a given task
- Evolving the ANNs by optimizing the weights and/or topology
- Mathematical optimization of RNNs is a hard task
- NE can be used to evolve RNNs efficiently

Neuro Evolution of Augmented Topology

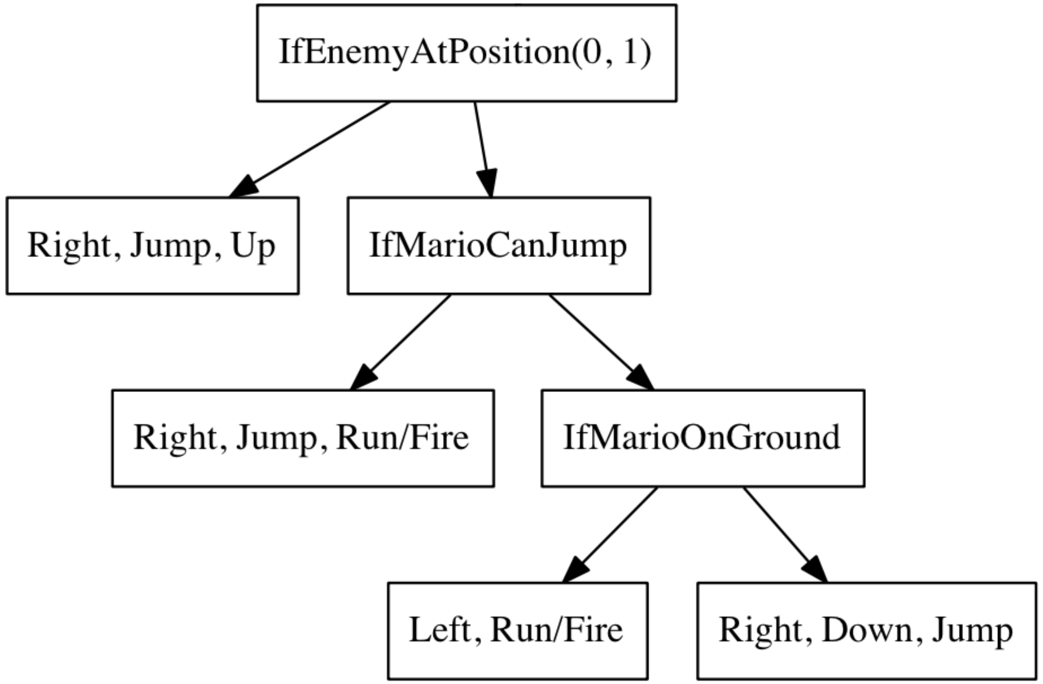
- Starting from a simple ANN
- Adding new nodes/connections and change the weights
- Speciation
- Enabling & Disabling connections
- Innovation Numbers

Neuro Evolution of Augmented Topology



Concept of NEAT [4]

Behavior Trees



Behavior Tree example [5]

- Encoding behavior of a controller
- Action Nodes:
  - Leafs
  - The final decision
- Condition Nodes:
  - If-else-statement
  - Branching nodes

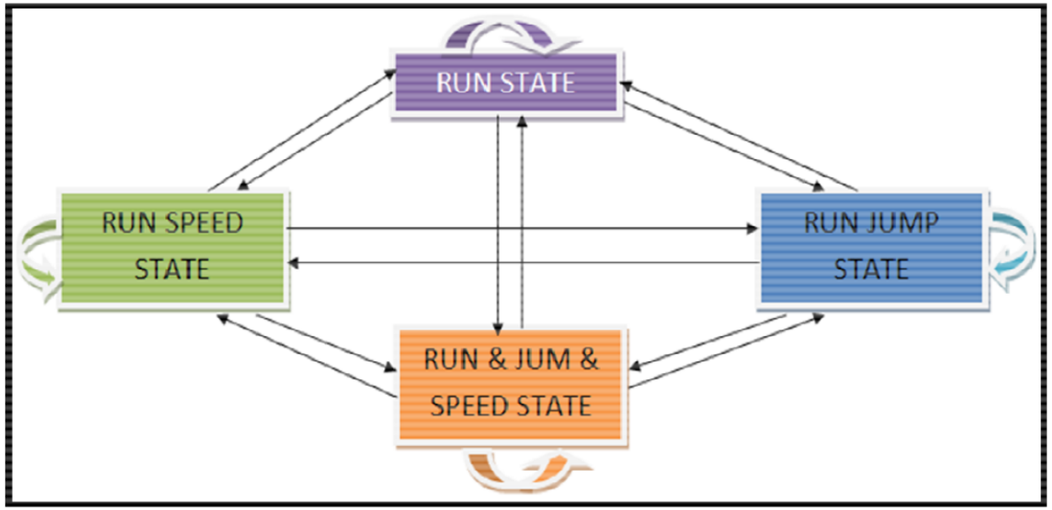
Evolving Behavior Trees

- BTs get encoded via a context-free gramatic into an array
- The array is used as a genome
- Crossover: Swapping subtrees of parents
- Mutation: Randomly replace nodes

Super Mario

Nintendo

Using GAs for Super Mario - FSM



State Machine [6]

Triggers

- Seen an enemy
- Seen an obstacle
- Seen nothing
- Seen enemy & seen hole
- Seen enemy & seen obstacle
- Seen hole & seen obstacle

Using GAs for Super Mario – Learning Levels

- A genome encodes a whole level
- The genome is somehow the key for a level
- Through Evolutionary Algorithm the genome is evolved

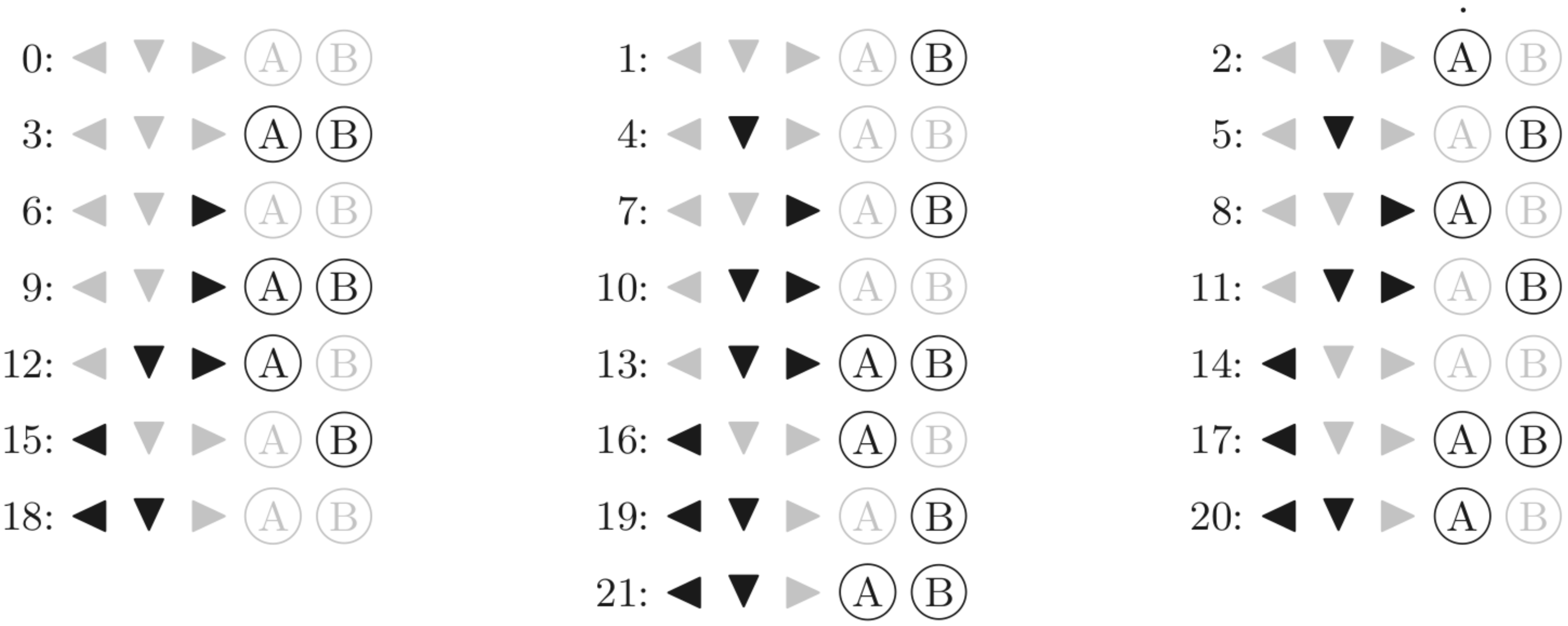
Using GAs for Super Mario – Learning Levels

- One game lasts for 200 seconds
- Discretized in 15 ticks → 3000 actions per game
- With 22 possible actions → 22^{3000} possible combinations

- Fitness: Distance + Killed Enemies + Collected Items

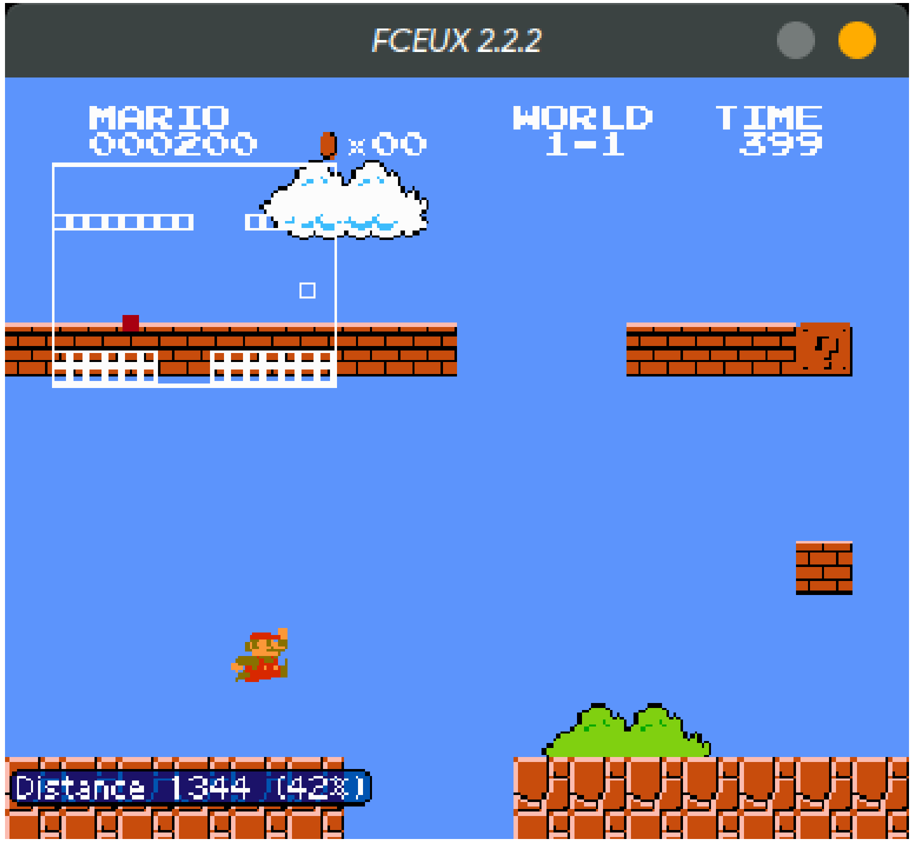
- Result: 12.000 points on average, 2010 Mario AI Championship Winner had 9000 points on average

Using GAs for Super Mario – Learning Levels



Combinations of Buttons [8]

Using GAs for Super Mario – NEAT



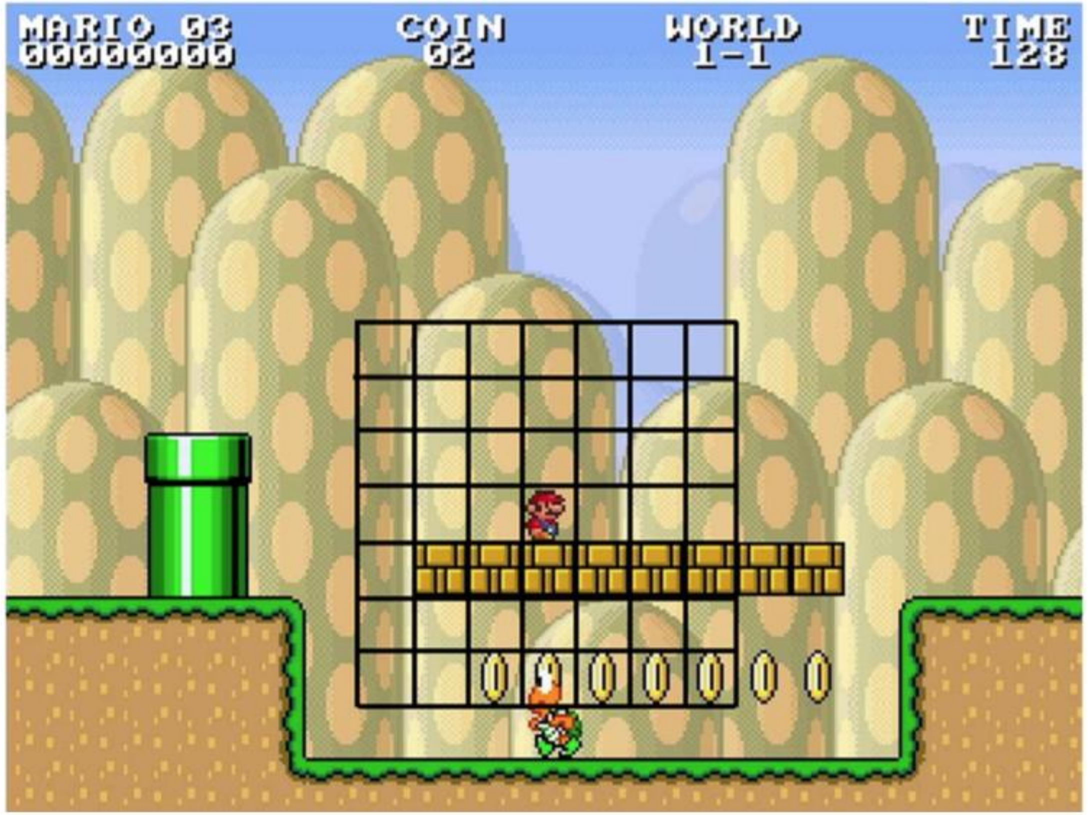
Super Mario learned with NEAT [8]

- Using NEAT to evolve a controller
- Input: 16x13 grid of view
- Output: 6 Buttons as Bit-Vector

→Controller were able to solve a level after 35 generations

Fitness: Distance

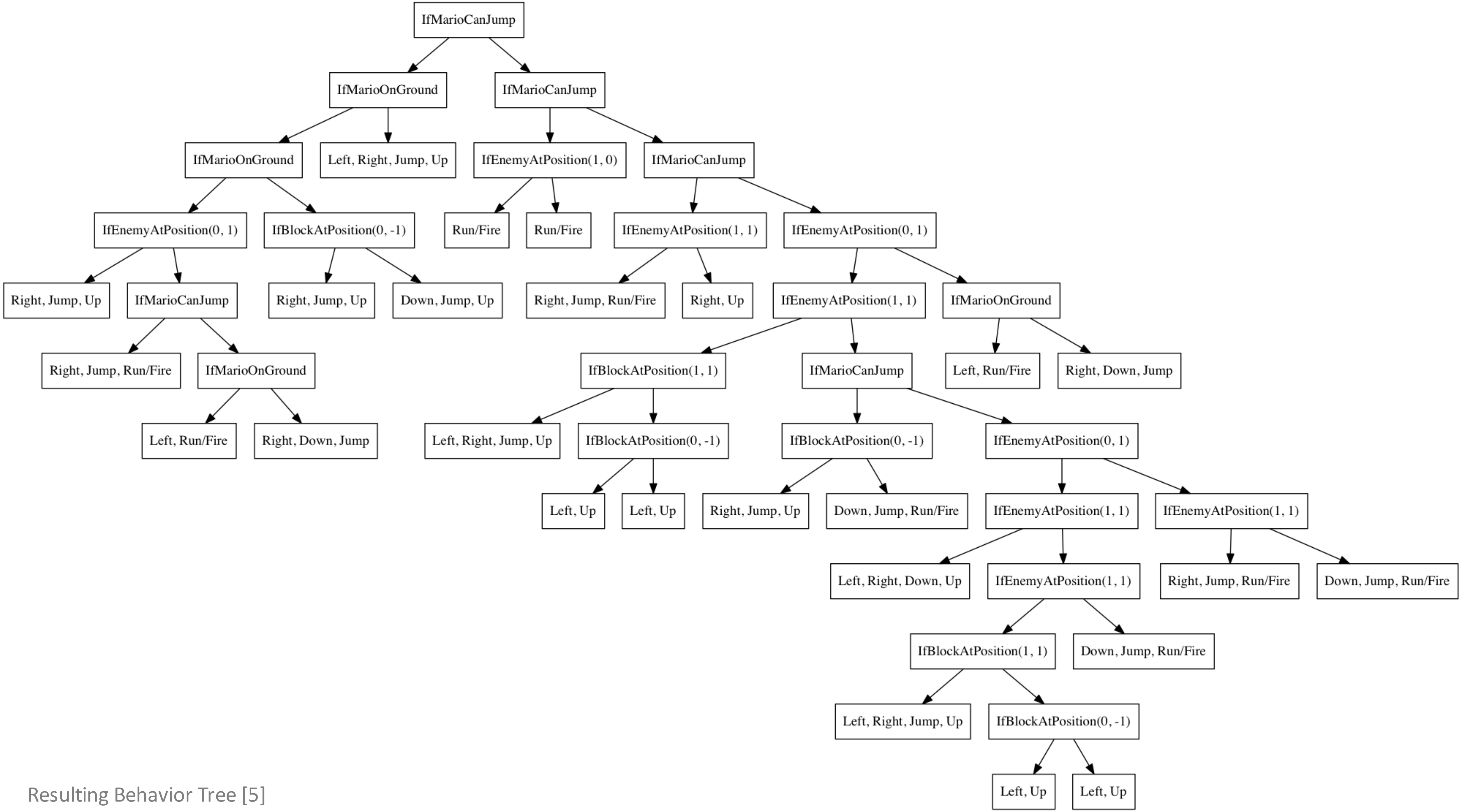
Using GAs for Super Mario – EBT



Super Mario Level with 7x7 input grid [5]

- Using a grid around Mario
- Entry can be enemie, block or empty
- Additional information:
  - Can Mario jump?
  - Is Mario on the ground?
- In the paper, they compared it to NEAT, using the grid as input

Fitness: Distance



Resulting Behavior Tree [5]

Conclusion

- Evolutionary Algorithms and Neuroevolution are a good approach for every Task where no perfect strategy is known
- GAs and NE can be used if a solution can be encoded as genome and a the performance of a solution can be rated
- GAs can find unusal solutions and are capable to cover a wide behavior diversity

- BUT: GAs need a lot of computing power and the parameters have to be optimized by hand in order to make the algorithm reach a good solution

References

[1] Casas, P.: https://www.r-bloggers.com/feature-selection-using-genetic-algorithms-in-r/
Visted: 21.05.19

[2] Nissen, V.: Einige Grundlagen Evolutionärer Algorithmen, 1998

[3] Pagliuca, P. et al: Maximizing adaptive power in Neuroevolution, 2018

[4] Stanley, K.; Miikkulainen, R.: Evolving Neural Networks through Augmenting Topologies, 2002

[5] Kuhn, K.; Foley, R.: A Comparison of Genetic Algorithms using Super Mario Bros, 2015

[6] Infinite Mario Bross AI using Genetic Algorithm

[7] Baldominos, A. et al: Learning Levels of Mario AI Using Genetic Algorithms, 2015

[8] Singhal, K. et al: Deep Reinforcement Learning in Mario, 2016