An Introduction to Neuroevolution

Paul Pauls,*Technical University of Munich (TUM)*

Advisor: Michael Adam,*Technical University of Munich (TUM)*

Evolutionary, Genetic and Neuroevolution Algorithms

Evolutionary Algorithms

„A generic population-based and meta-heuristically optimized algorithmic solution to an applied problem“

(Source: [16])

Evolutionary Algorithms

„A generic population-based and meta-heuristically optimized algorithmic solution to an applied problem“

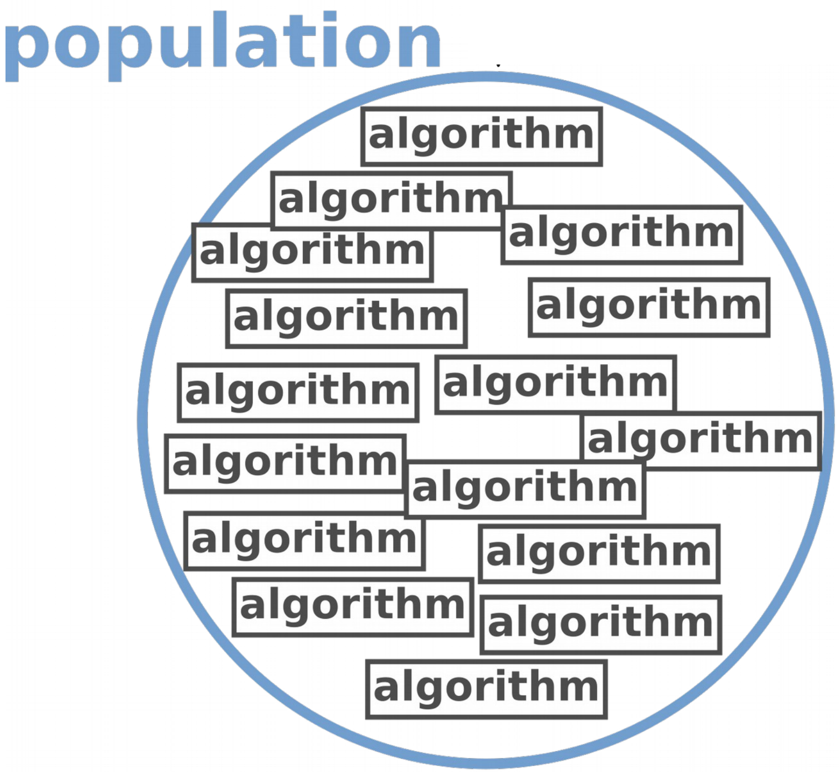
(Source: [16])

Evolutionary Algorithms

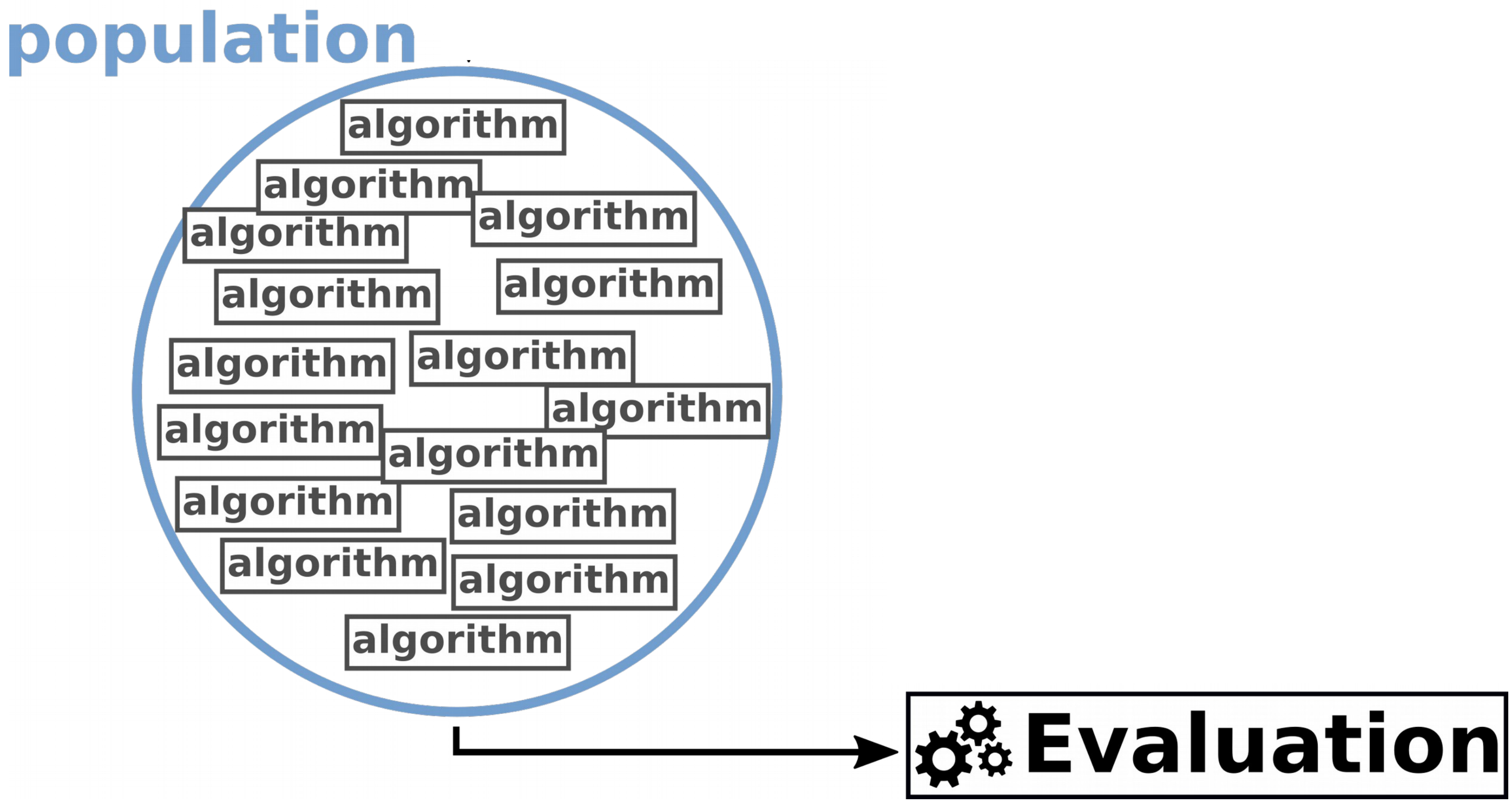
„A generic population-based and meta-heuristically optimized algorithmic solution to an applied problem“

(Source: [16])

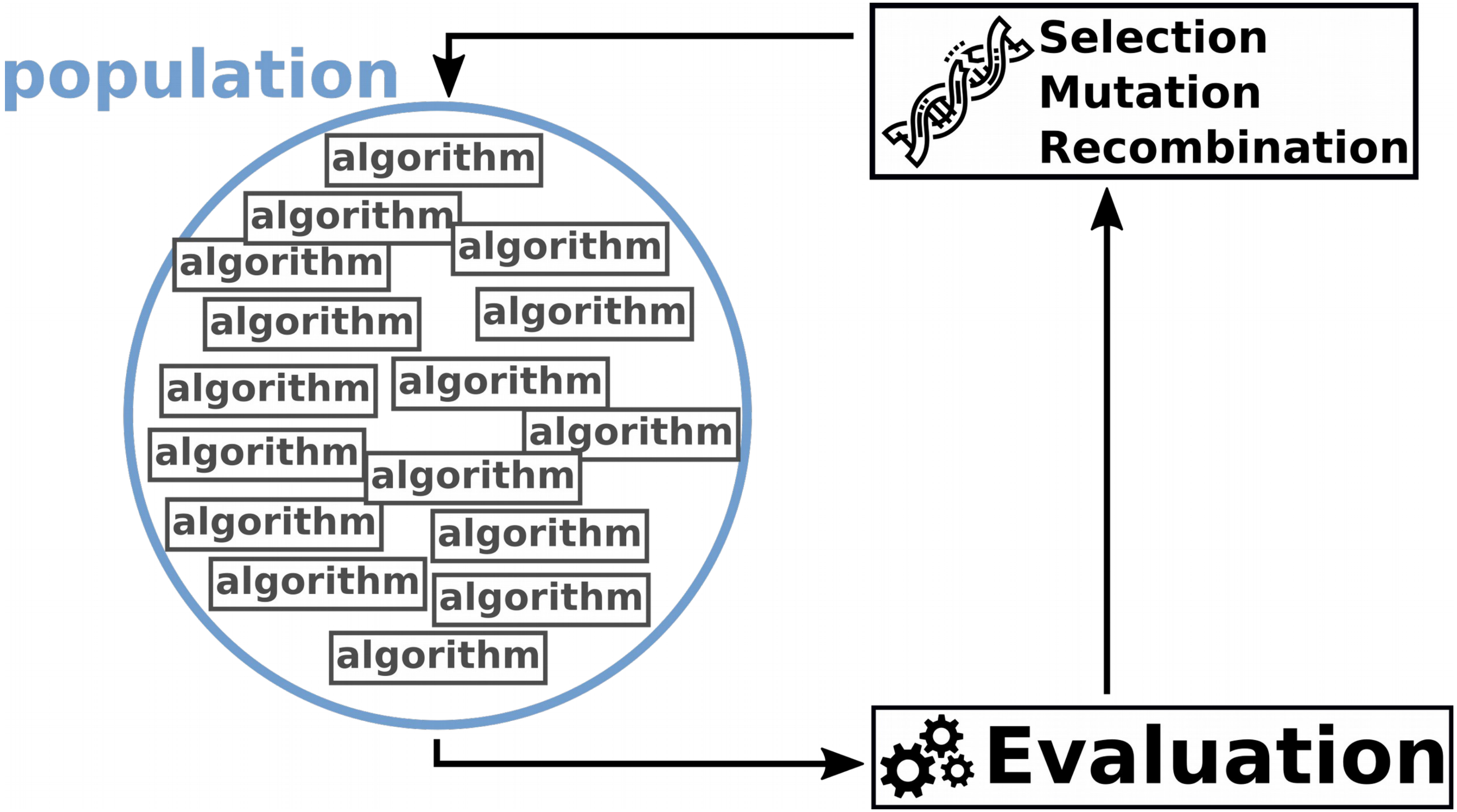
Evolutionary Algorithms



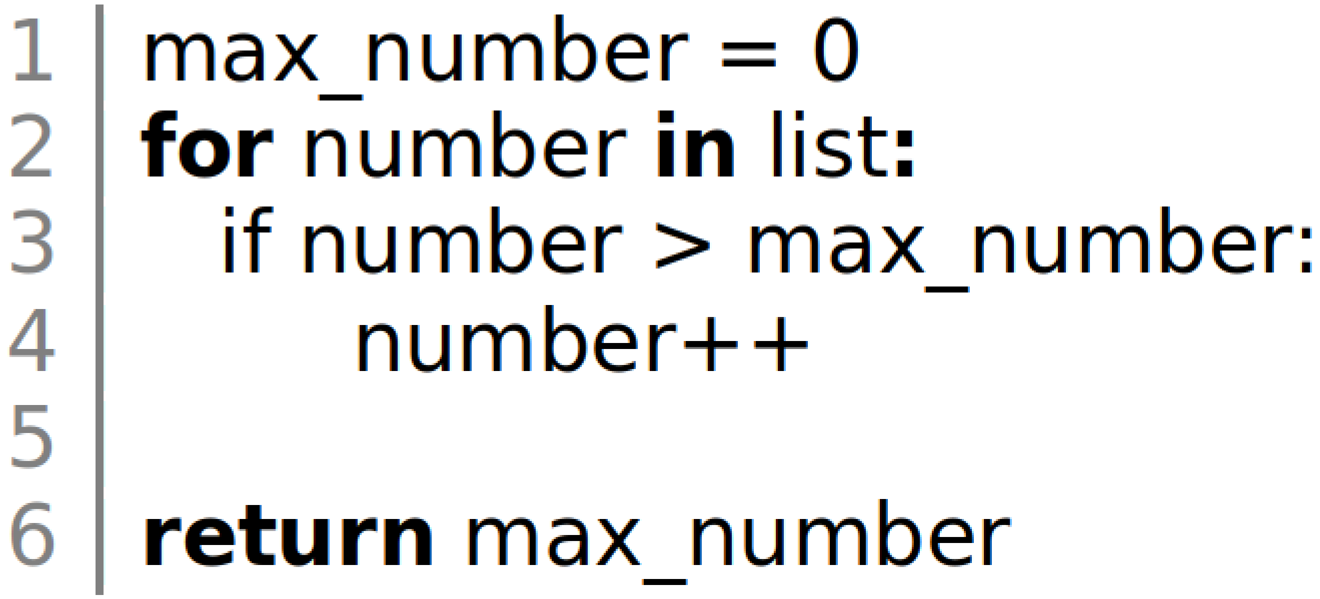
Evolutionary Algorithms



Evolutionary Algorithms

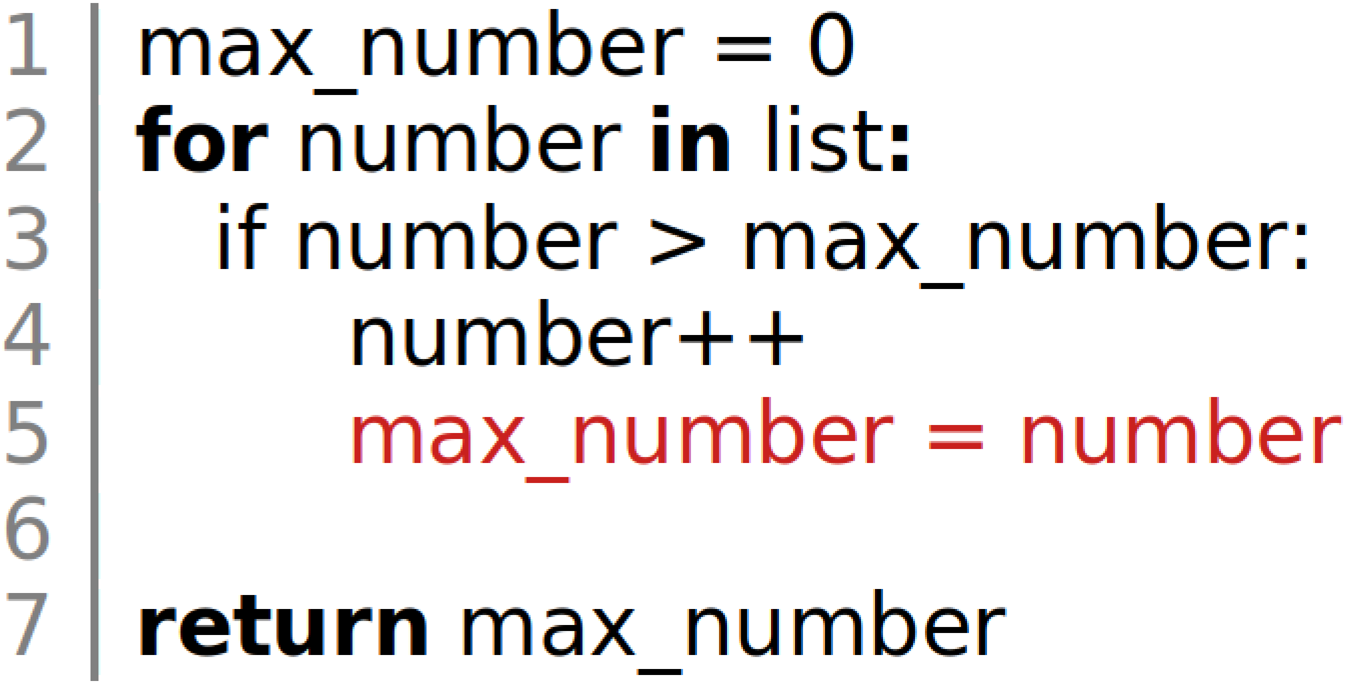


Evolutionary Algorithms – Mutation



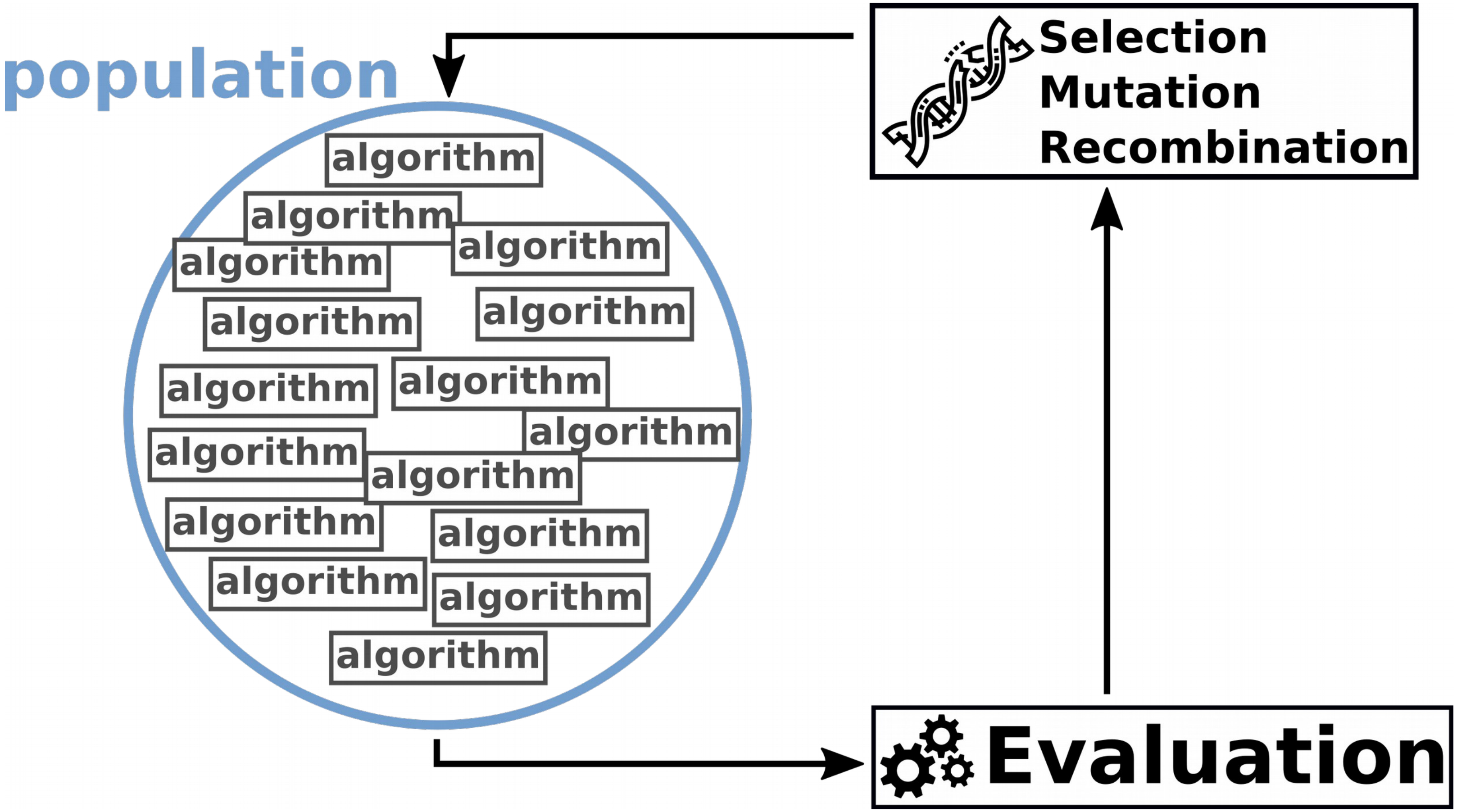
```
max_number = 0
for number in list:
    if number > max_number:
        number++

return max_number
```

Evolutionary Algorithms – Mutation



```
max_number = 0
for number in list:
    if number > max_number:
        number++
        max_number = number

return max_number
```

Evolutionary Algorithms

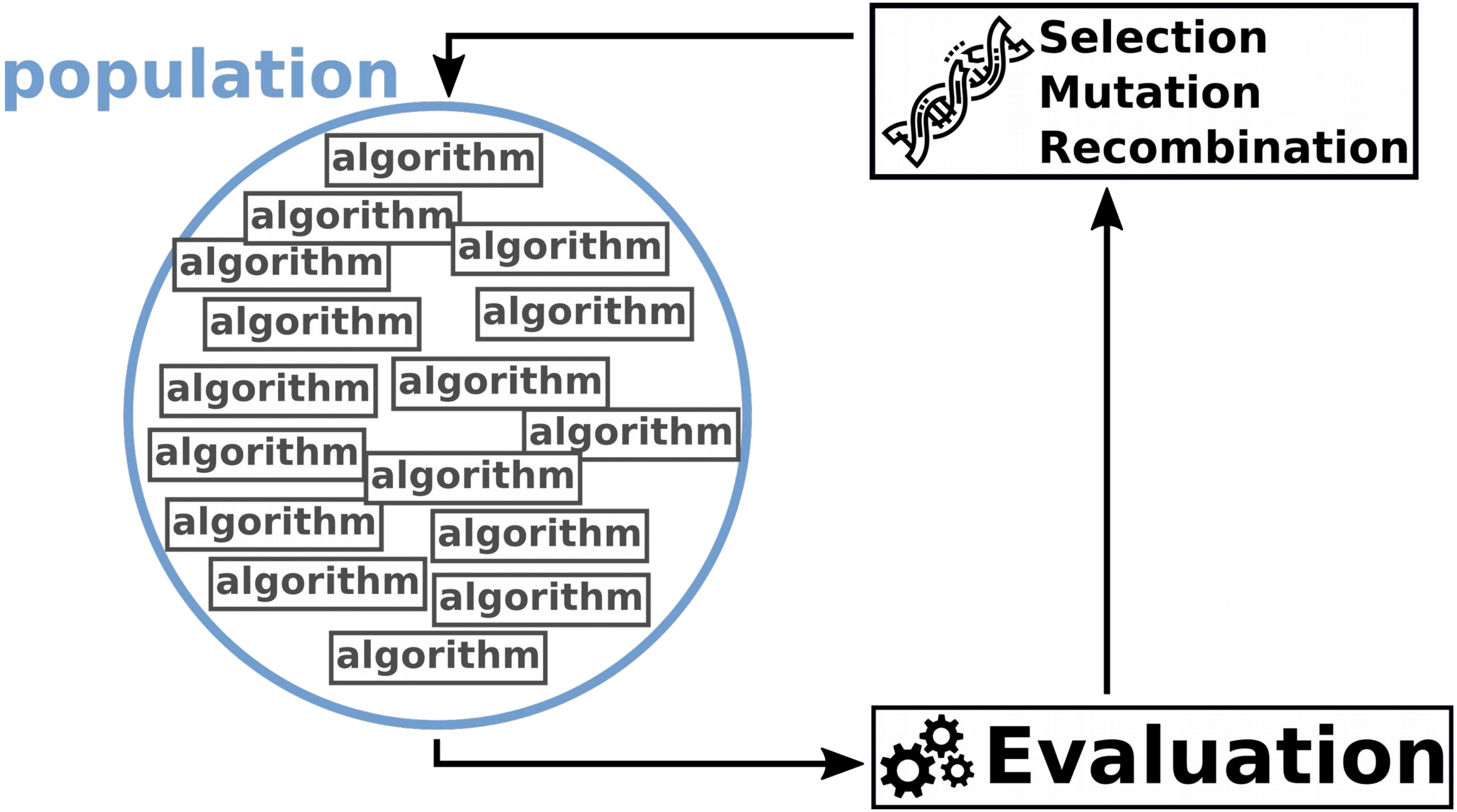


Evolutionary Algorithms

„A generic population-based and meta-heuristically optimized algorithmic solution to an applied problem“

(Source: [16])

Evolutionary Algorithms

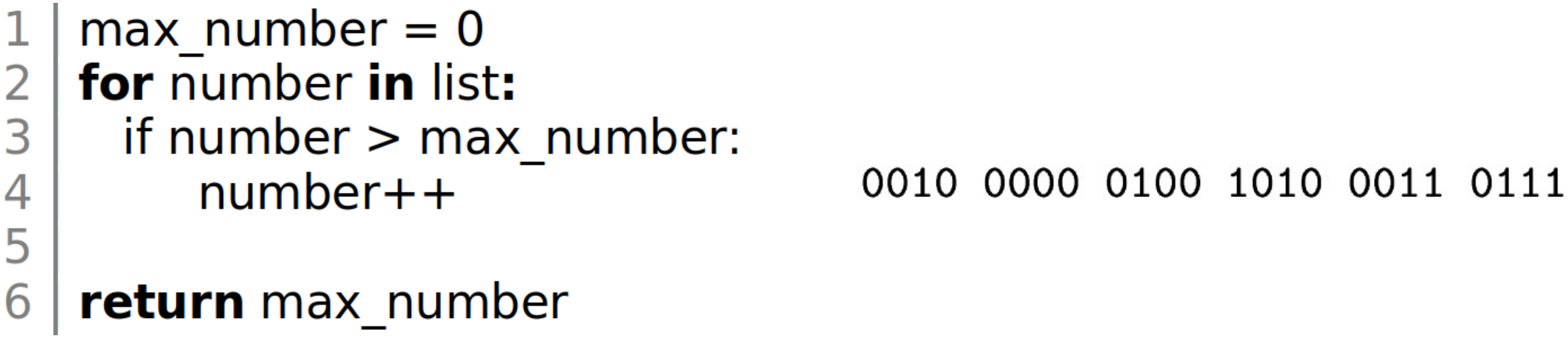


Evolutionary Algorithms

„A generic population-based and meta-heuristically optimized algorithmic solution to an applied problem“

(Source: [16])

Genetic Algorithms



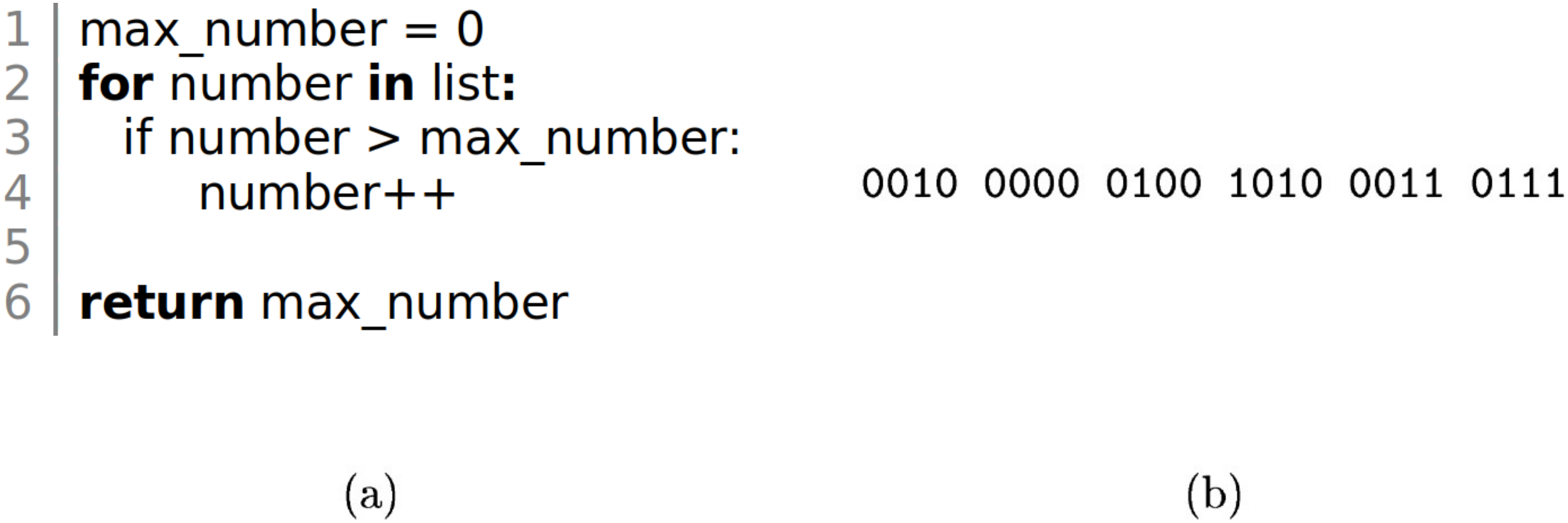
```
max_number = 0
for number in list:
   if number > max_number:
         number++

return max_number
```

(a)

```
0010 0000 0100 1010 0011 0111
```

(b)

Genetic Algorithms



```
max_number = 0
for number in list:
  if number > max_number:
      number++

return max_number
```

(a)

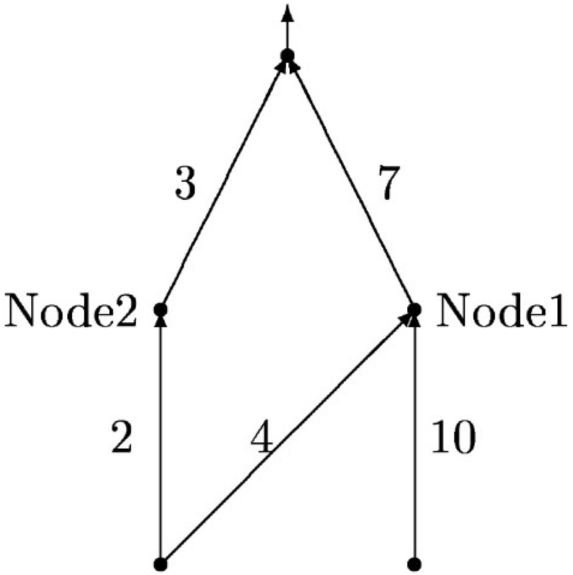
```
0010 0000 0100 1010 0011 0111
```

(b)

[phenotype]

[genotype]

Genetic Algorithms



(a)

[phenotype]

0010 0000 0100 1010 0011 0111

(b)

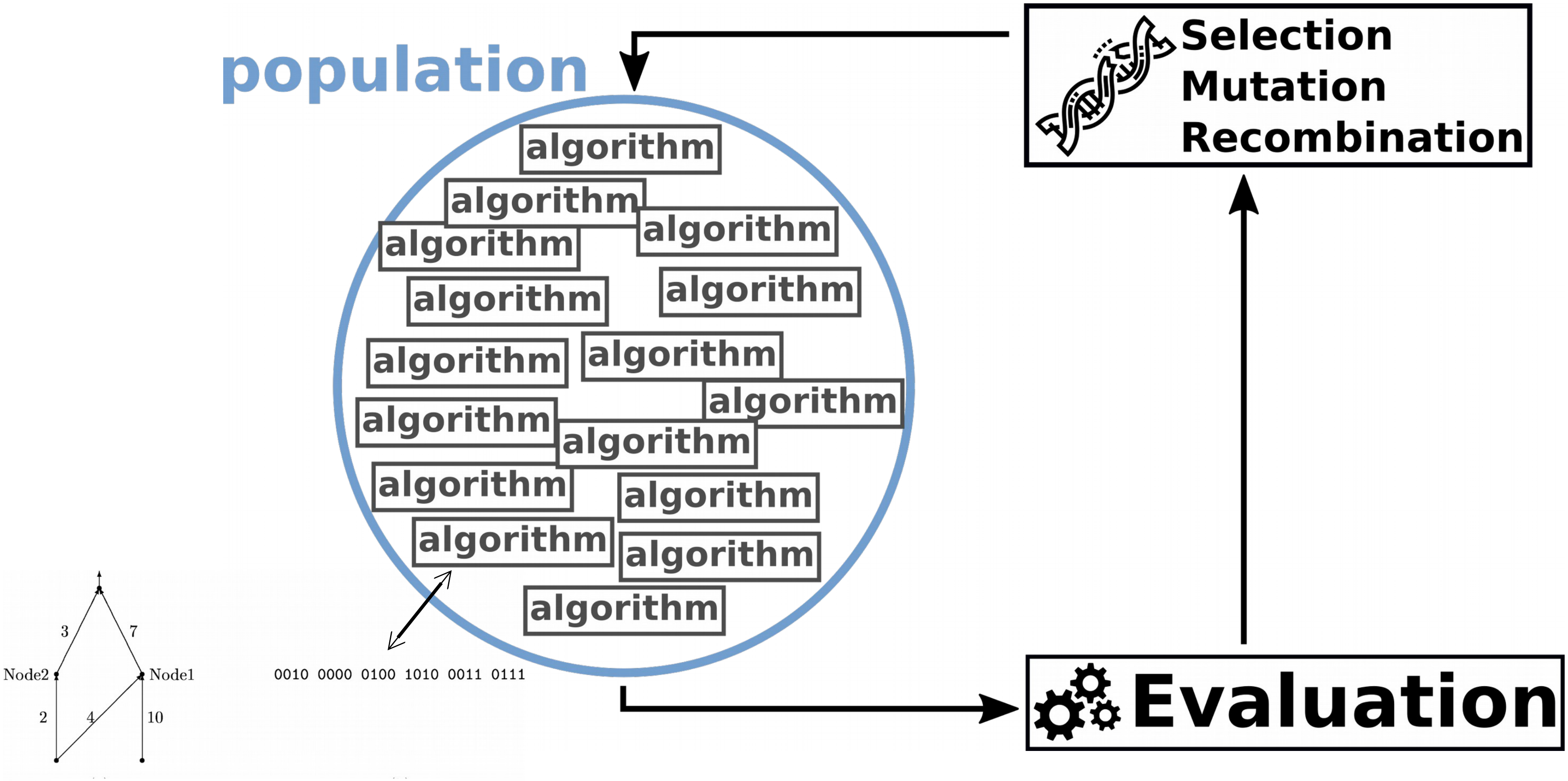
[genotype]

Source: [4]

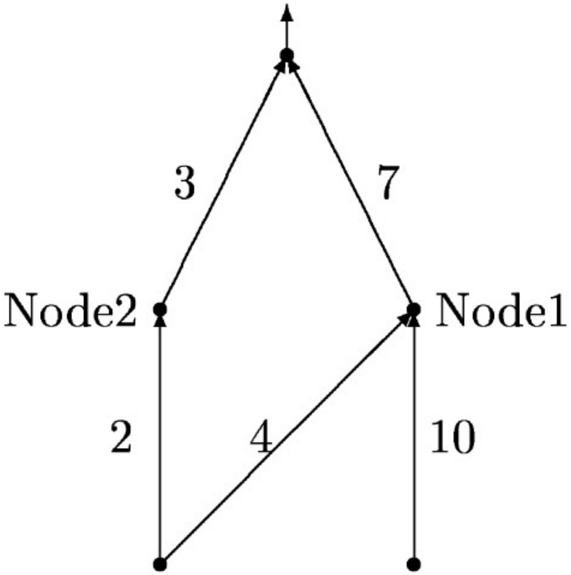
Neuroevolution

„A *neuroevolution algorithm* is a *genetic algorithm*, whose search-space (genotypes) consists **only** of artificial neural networks. A *genetic algorithm* in turn is an *evolutionary algorithm* that evolves genotypes through **mutation, recombination and selection.**“

Neuroevolution



Neuroevolution – Mutation



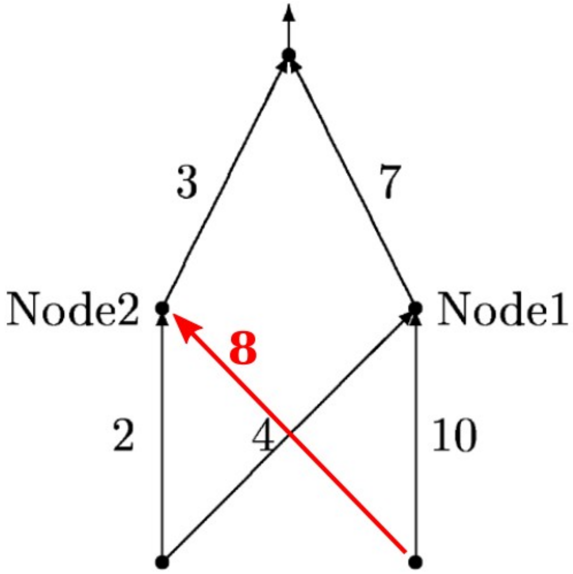
(a)

0010 0000 0100 1010 0011 0111

(b)

Source: [4]

Neuroevolution – Mutation

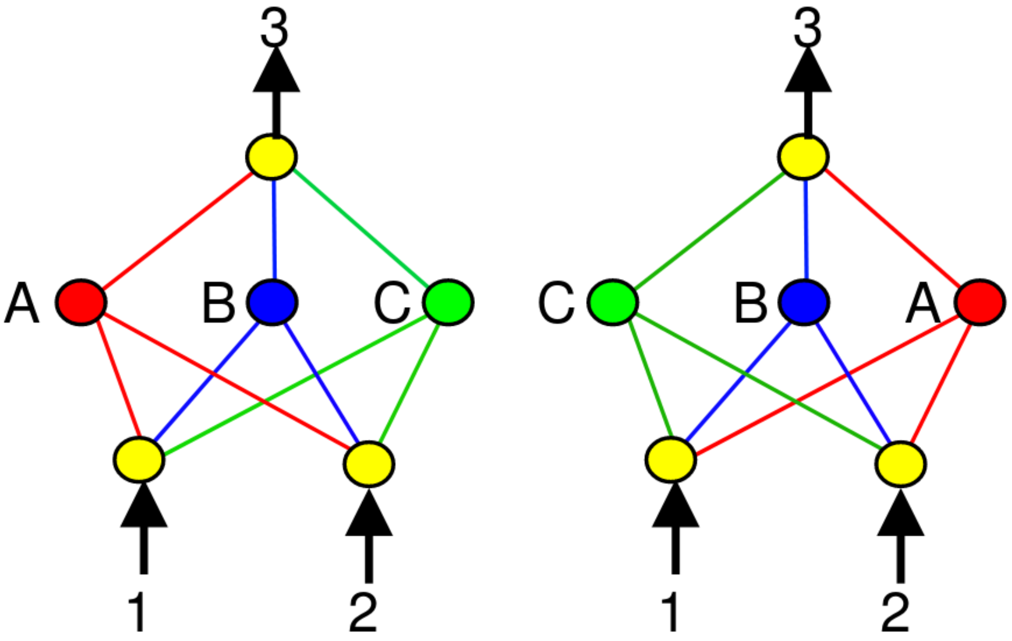


(a)

0010 **1**000 0100 1010 0011 0111

(b)

Neuroevolution – Recombination



[A,B,C]
X[C,B,A]

Crossovers: [A,B,A] [C,B,C]
(both are missing information)

Source: [7]

Neuroevolution – Selection

Periodical **removal** of the **lowest performing algorithms** in the population

Neuroevolution

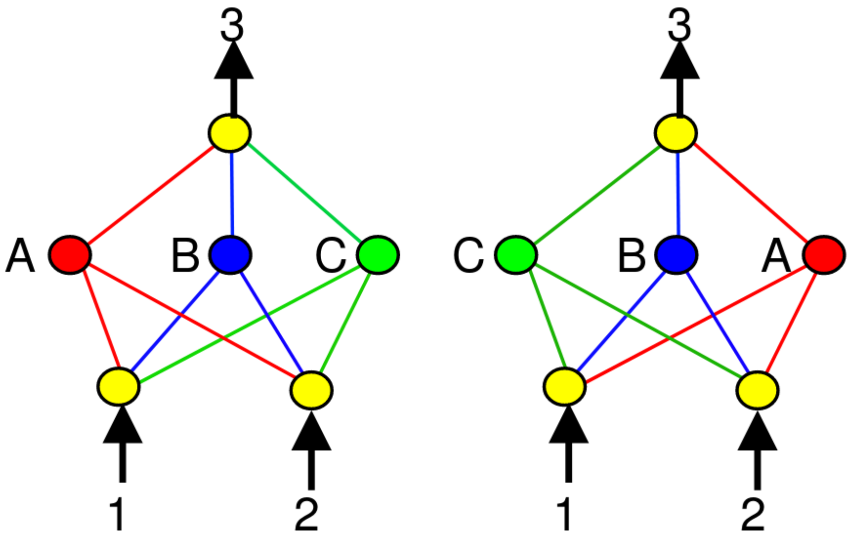
> „Neuroevolution is the - possibly boundless - process in which by the means of a genetic algorithm the population of artificial neural networks is increasingly optimized in order to maximize the accuracy or fitness of the best ANN.“

Neuroevolution of Augmenting Topologies (abbr. NEAT)

Neuroevolution of Augmenting Topologies

- Published in 2002 by Kenneth O.Stanley and Risto Miikkulainen [6]
- Outperformed all contemporary neuroevolution systems when introduced [8]
- Achieved performance because it solved a fundamental problem in neuroevolution, while still being elegantly simple [8, chap. 3.2]
- Still one of the most prominent neuroevolution systems today and considered a benchmark in the field [22]

NEAT – Principle of Historical Markings

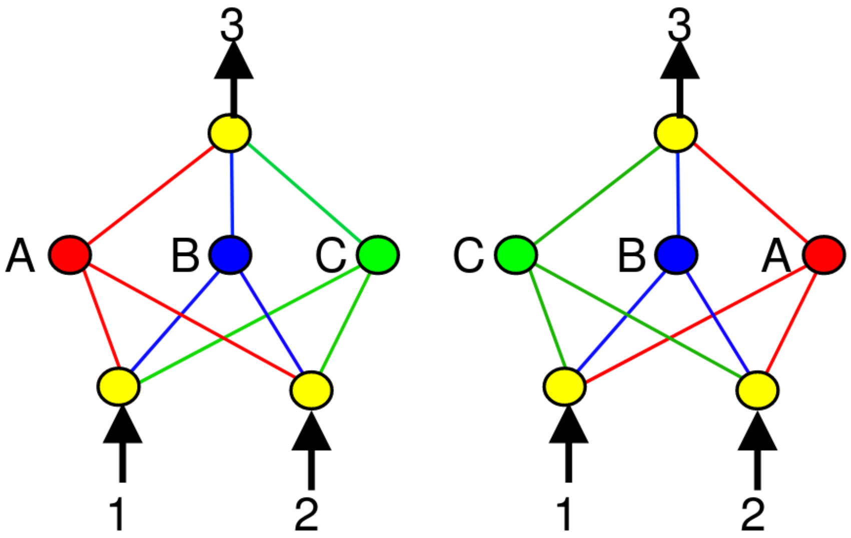


[A,B,C]
X[C,B,A]

Crossovers: [A,B,A] [C,B,C]
(both are missing information)

Source: [7]

NEAT – Principle of Historical Markings



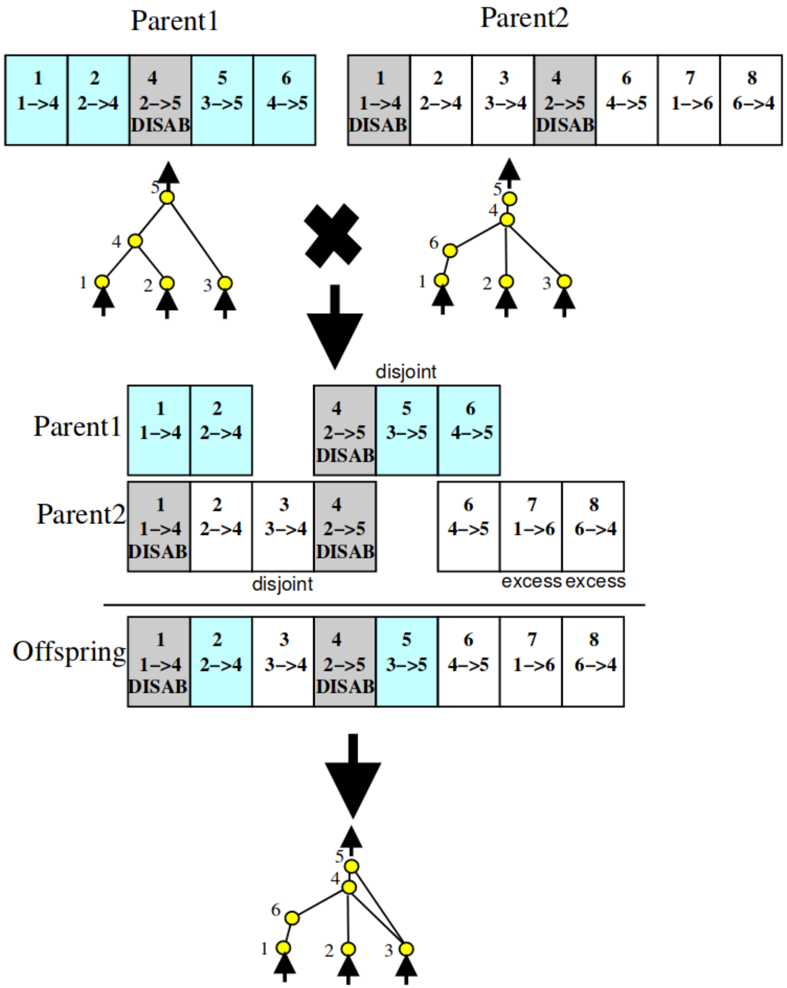
[A,B,C]
X [C,B,A]

Crossovers: [A,B,A] [C,B,C]
(both are missing information)

Source: [7]

Principle: „Enable information-preserving crossovers by keeping track of genes – and their presence in genomes – by introducing ***innovation numbers***“

NEAT – Principle of Historical Markings



Parent1

1 1->4	2 2->4	4 2->5 DISAB	5 3->5	6 4->5

Parent2

1 1->4 DISAB	2 2->4	3 3->4	4 2->5 DISAB	6 4->5	7 1->6	8 6->4

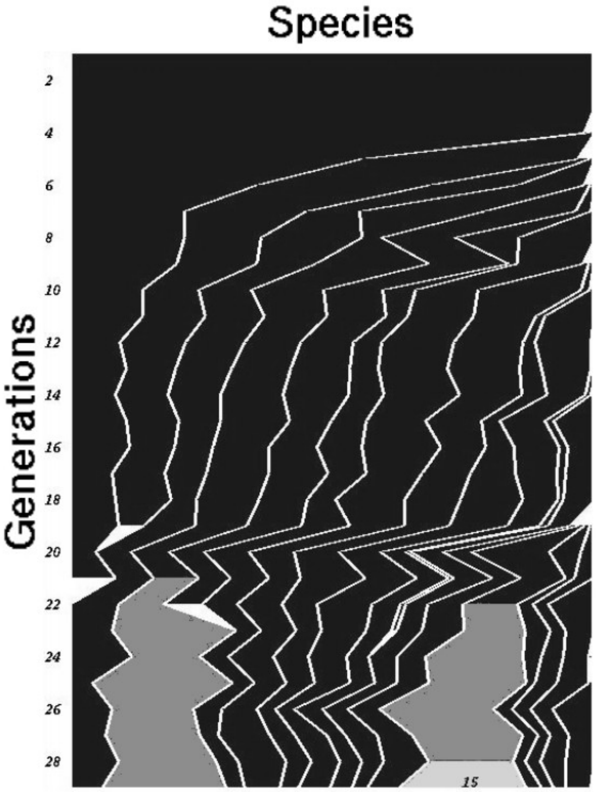
					disjoint			
Parent1	1 1->4	2 2->4		4 2->5 DISAB	5 3->5	6 4->5		
Parent2	1 1->4 DISAB	2 2->4	3 3->4	4 2->5 DISAB		6 4->5	7 1->6	8 6->4
			disjoint				excess	excess
Offspring	1 1->4 DISAB	2 2->4	3 3->4	4 2->5 DISAB	5 3->5	6 4->5	7 1->6	8 6->4

Source: [7]

NEAT – Principle of Speciation

Principle: „Protect innovation by dividing population into niches according to their distinctiveness and only comparing niches against one another“

NEAT – Principle of Speciation

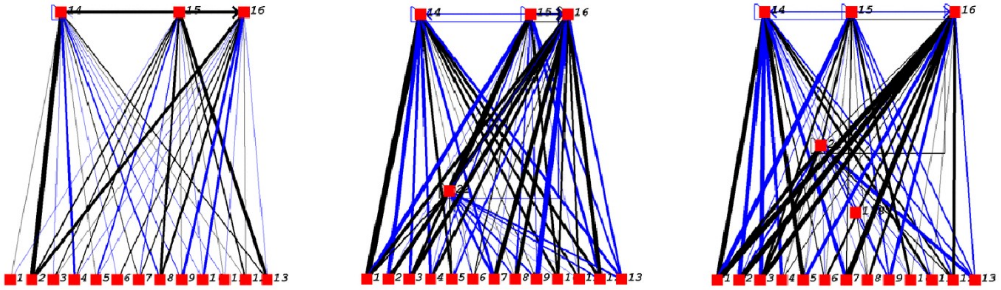


Source: [8]

Principle: „Protect innovation by dividing population into niches according to their distinctiveness and only comparing niches against one another“

NEAT – Principle of Complexification

- Explain and vizualize this principle



(a) Generation 100 (b) Generation 267 (c) Generation 315

Figure 5.4: **Complexification of a winning species.** The best networks in the same species are shown at landmark generations. Nodes are depicted as squares beside their node numbers, and line thickness represents the strength of connections. Over time, the networks became more complex and gained skills. (a) The champion from generation 10 had no hidden nodes. (b) The addition of h_{22} and its respective connections gave new abilities. (c) The appearance of h_{172} refined existing behaviors.

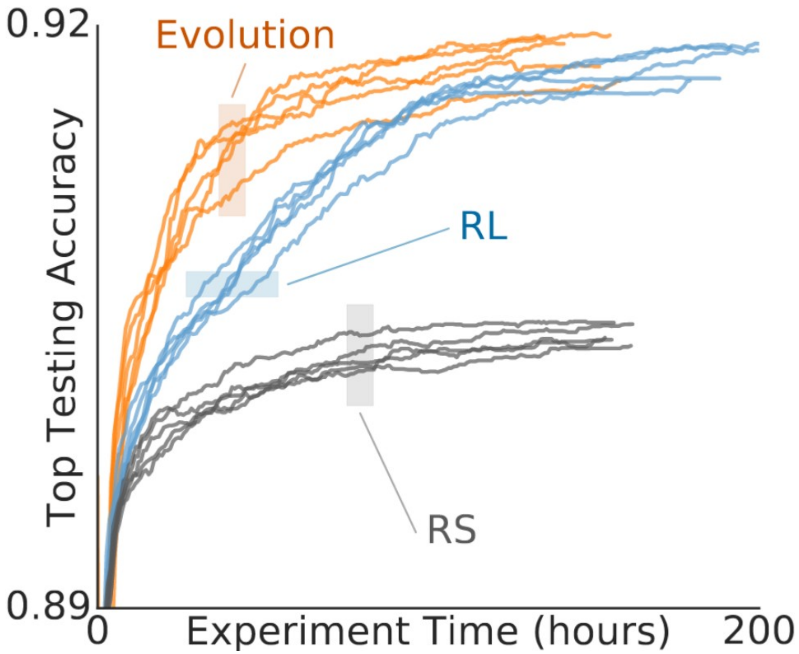
Neuroevolution of Augmenting Topologies

- Give Summary and Overview

Neuroevolution Performance and Practical Applications

Neuroevolution – Performance

- Give Performance Review of NEAT back in 02 and NE in todays real research



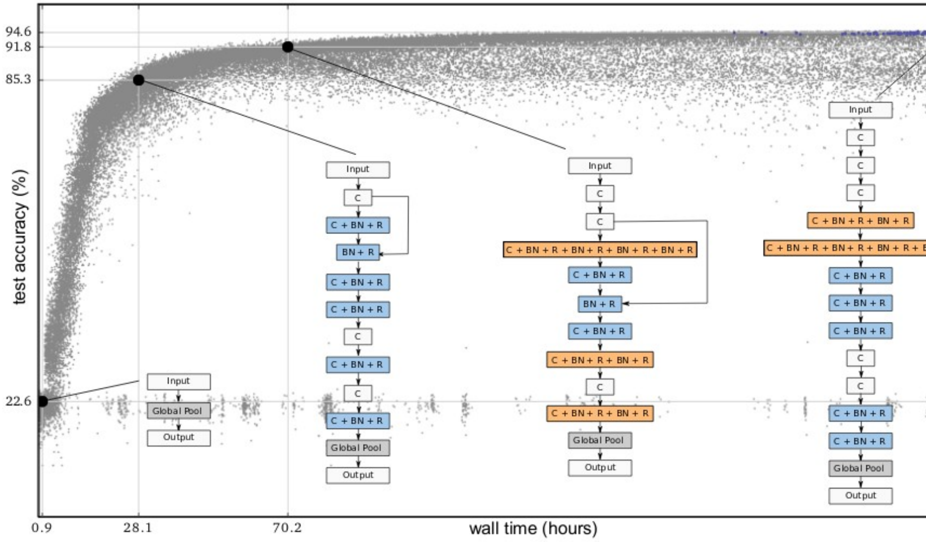


Figure 1. Progress of an evolution experiment. Each dot represents an individual in the population. Blue dots (darker, top-r
The rest have been killed. The four diagrams show examples of discovered architectures. These correspond to the best ind

Neuroevolution – Practical Examples

- Don‘t go much into detail, but mention that there are no common NE frameworks, though there currently is work on that.
- Blend over to live example

An Introduction to Neuroevolution

Paul Pauls,*Technical University of Munich (TUM)*

Advisor: Michael Adam,*Technical University of Munich (TUM)*

References

[1] Turing - Computing Machinery and Intelligence; Oct. 1950; https://academic.oup.com/mind/article/LIX/236/433/986238

[2] Anderson - Learning to control an inverted pendulum using neural networks; 1989; https://ieeexplore.ieee.org/document/24809

[3] Gruau, Bernard-lyon, et al - Neural Network Synthesis Using Cellular Encoding And The Genetic Algorithm; 1994; https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.29.5939

[4] Yao - Evolving Artificial Neural Networks; 1999; http://avellano.fis.usal.es/∼lalonso/compt_soft/articulos/yao99evolving.pdf

[5] Gomez, Miikkulainen - Solving Non-Markovian Control Tasks with Neuroevolution; 1999; http://nn.cs.utexas.edu/downloads/papers/gomez.ijcai99.pdf

[6] Stanley, Miikkulainen - Evolving Neural Networks through Augmented Topologies; 2002; http://nn.cs.utexas.edu/downloads/papers/stanley.ec02.pdf

[7] Stanley, Miikkulainen - Efficient Evolution of Neural Network Topologies; 2002; http://nn.cs.utexas.edu/downloads/papers/stanley.cec02.pdf

[8] Stanley - Efficient Evolution of Neural Networks through Complexification; Aug 2004; http://nn.cs.utexas.edu/downloads/papers/stanley.phd04.pdf

[9] Stanley, D'Ambrosio, et al - A Hypercube-Based Indirect Encoding for Evolving Large-Scale Neural Networks; 2009; http://axon.cs.byu.edu/∼dan/778/papers/NeuroEvolution/stanley3**.pdf

[10] Aaltonen, Adelman, et al - Measurement of the top-quark mass with dilepton events selected using neuroevolution at CDF; Apr 2009; https://www.ncbi.nlm.nih.gov/pubmed/19518620

[11] Risi, Stanley - Enhancing ES-HyperNEAT to Evolve More Complex Regular Neural Networks; Jul 2011; http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.365.4332

[12] Lehman, Stanley - Novelty Search and the Problem with Objectives; Oct 2011; https://www.cs.ucf.edu/eplex/papers/lehman_gptp11.pdf

[13] Hausknecht - A Neuroevolution Approach to General Atari Game Playing; 2012; https://www.cs.utexas.edu/∼mhauskn/projects/atari/movies.html

[14] Ciresan, Giusti, et al - Deep Neural Networks Segment Neuronal Membranes in Electron Microscopy Images; 2012; http://people.idsia.ch/∼juergen/nips2012.pdf

[15] Krizhevsky, Sutskever, et al - ImageNet Classification with Deep Convolutional Neural Networks; 2012; https://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

[16] Holland - Scholarpedia Article on 'Genetic Algorithms'; Oct 2012; http://www.scholarpedia.org/article/Genetic_algorithms

[17] Mnih, Kavukcuoglu, et al - Playing Atari with Deep Reinforcement Learning; Dec 2013; https://arxiv.org/abs/1312.5602

[18] Schmidhuber - Deep Learning in Neural Networks; Apr 2014; https://arxiv.org/abs/1404.7828

[19] Cully, Clune, et al - Robots that can adapt like animals; May 2015; https://www.nature.com/articles/nature14422

[20] Miikkulainen, Liang, et al - Evolving Deep Neural Networks; Mar 2017; https://arxiv.org/abs/1703.00548

[21] Franca - Neuroevolution of Augmenting Topologies Applied to the Detection of Cancer in Medical Images; Feb 2018; http://www.bcc.ufrpe.br/sites/www.bcc.ufrpe.br/files/Luiz%20Fran%C3%A7a.pdf

[22] Frolov - Neuroevolution: A Primer on Evolving Artificial Neural Networks; Oct 2018; https://www.inovex.de/blog/neuroevolution/

[23] Real, Aggarwal, et al - Regularized Evolution for Image Classifier Architecture Search; Feb 2019; https://arxiv.org/abs/1802.01548

[24] CodeReclaimers - NEAT Python; Jun 2019; https://github.com/codereclaimers/neat-python

[25] NEAT Software Catalog; Jun 2019; http://eplex.cs.ucf.edu/neat_software/

[26] Pauls - SuperMario World NEAT Agent; Jun 2019; https://github.com/PaulPauls/SuperMarioWorld-NEAT-Agent

[27] Tensorflow 2.0 Beta; Jun 2019; https://www.tensorflow.org/beta

[28] Tensorflow addons; Jun 2019; https://github.com/tensorflow/addons

[29] Pauls - Tensorflow Neuroevolution; Jun 2019; https://github.com/PaulPauls/Tensorflow-Neuroevolution