Organic Computing

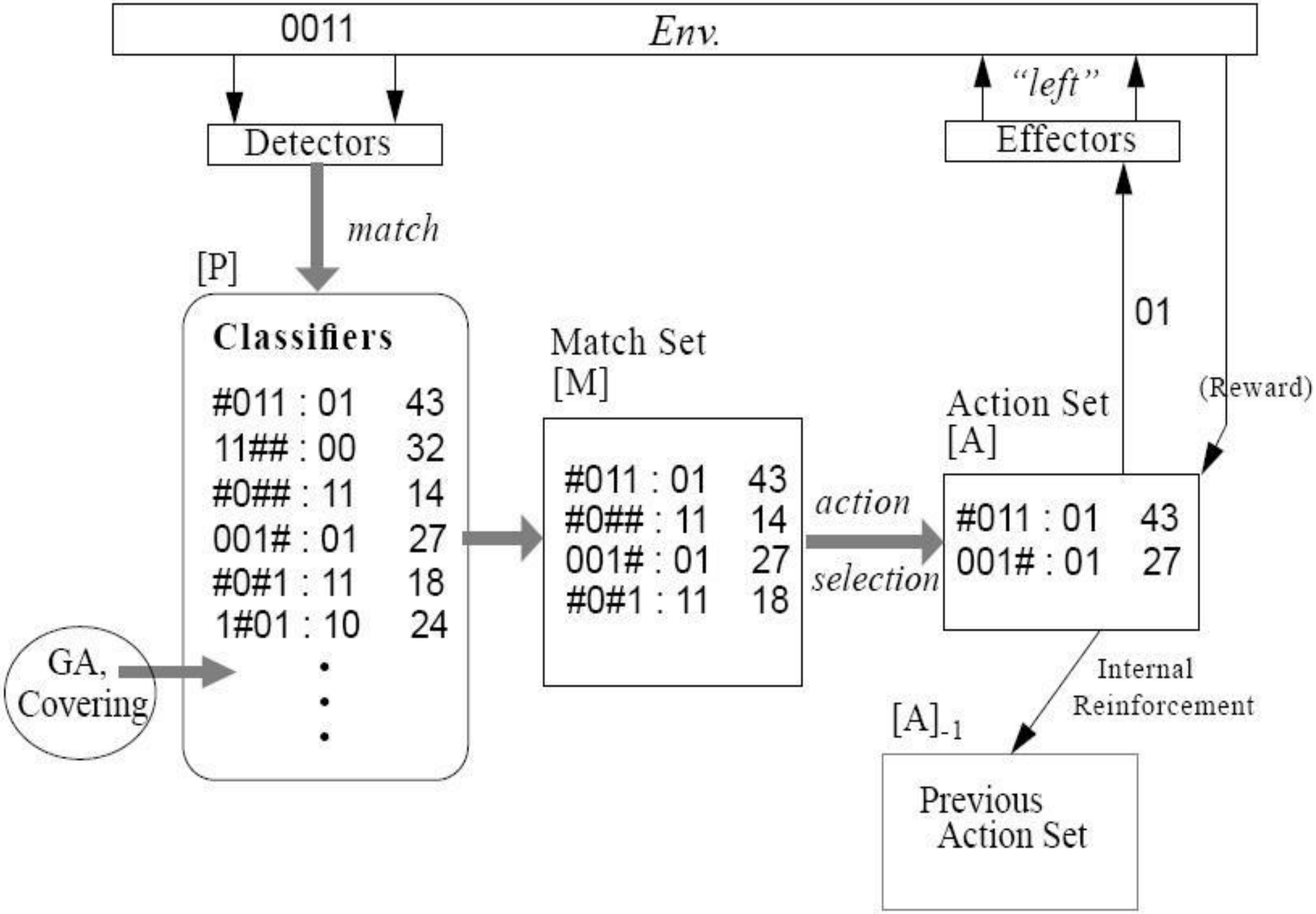
Dr. Sebastian von Mammen

Summer Term 2015

Part 2: Learning

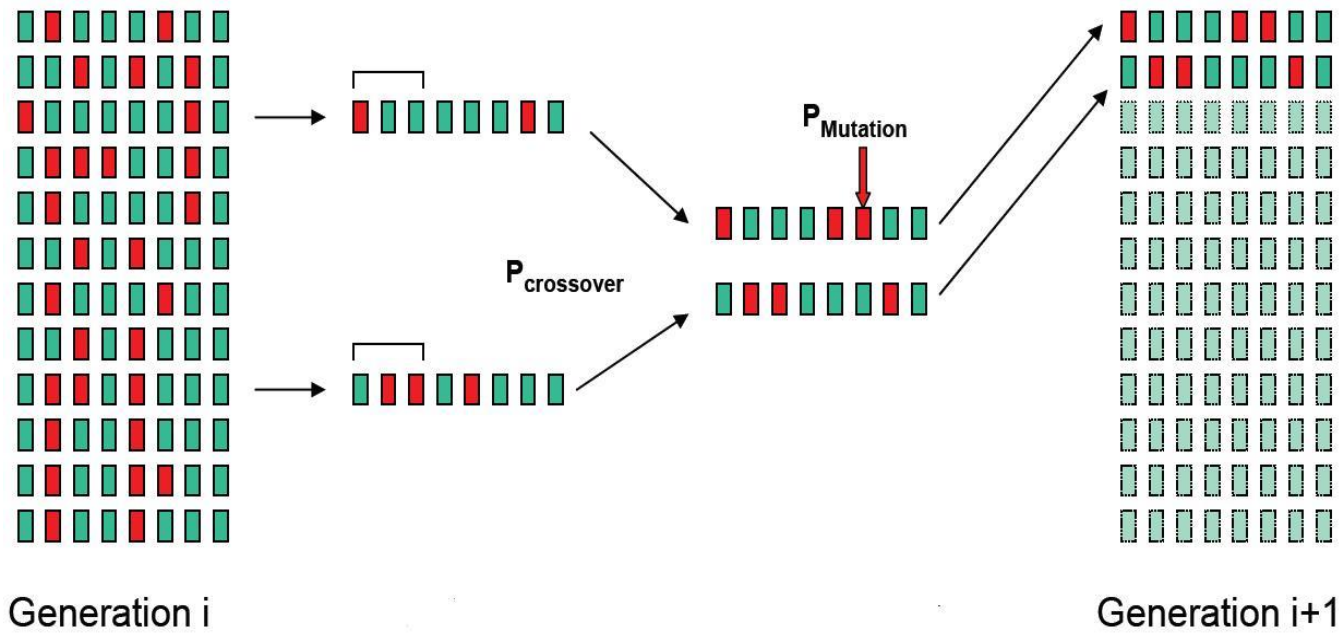
Slides are prepared in cooperation with
Prof. Müller-Schloer, Leibniz Universität Hannover

ZCS - Overview



- inspired by Darwinian evolution:

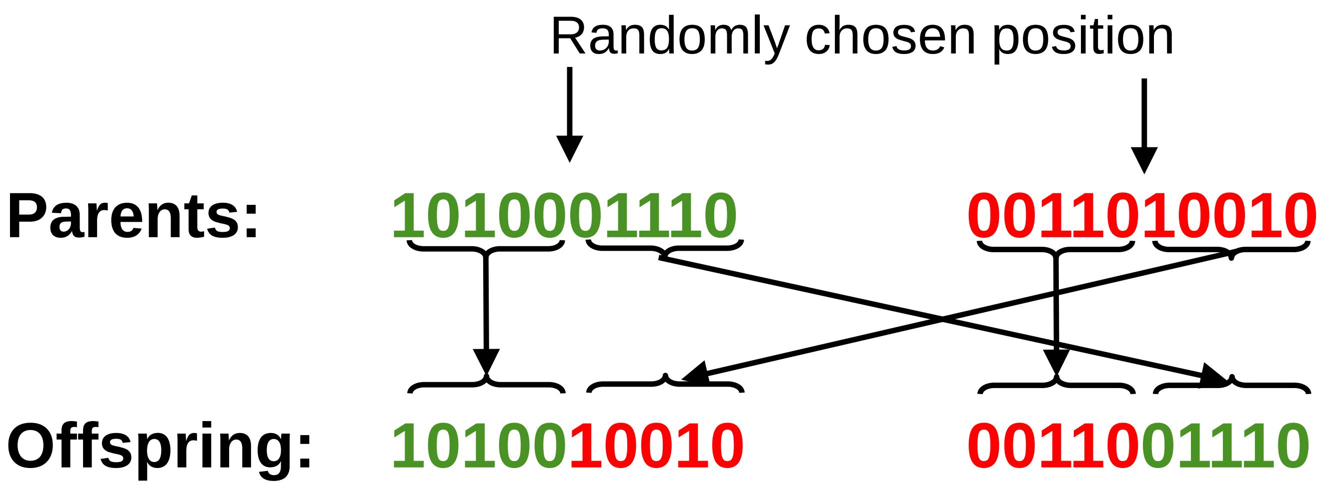
  "survival of the fittest"
- at each invocation two classifiers are selected as parents: roulette-wheel selection based on fitness
- with fixed probabilities, these "parents" are subject to "genetic" operations
  - crossover (recombination)
  - mutation
- resulting two classifiers get half of their parents' fitness
- rate at which GA is invoked is application dependent
  - too frequent: noisy fitness
  - too seldom: slow development
  - classifiers should have been evaluated a "couple of times" before becoming parents



- one-point crossover in the example: 3/5 bits
- multi-point crossover: more than one "cut point" selected
- mutation: randomly flips a bit (typically very small probability)

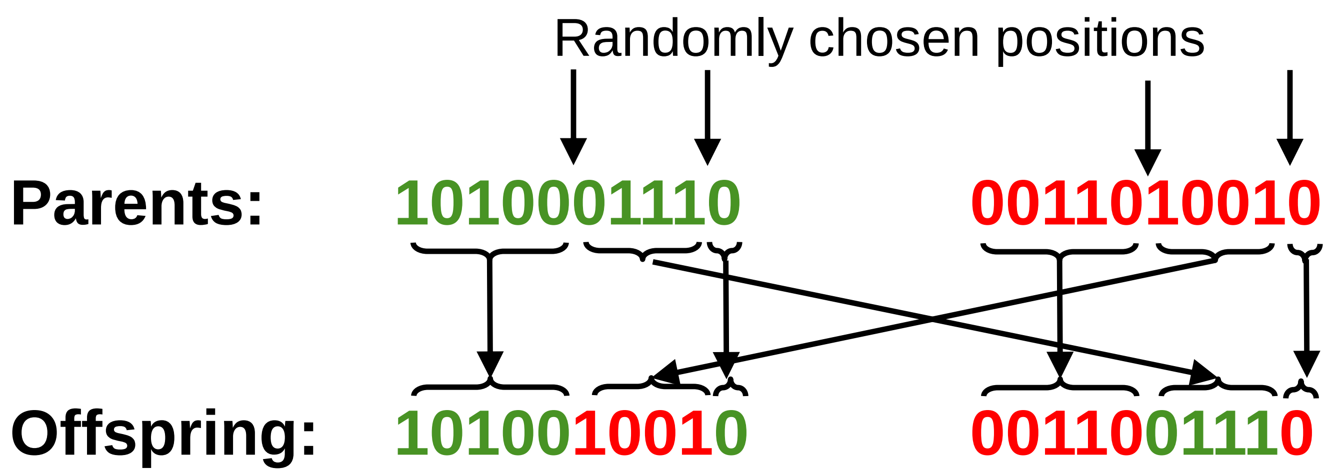
Genetic operators: 1-point crossover

- one position in the chromosomes is chosen randomly
- recombining the parents' genetic material ahead and after this position yields two offspring



Genetic operators: 2-point crossover

- two positions in the chromosomes are chosen randomly
- avoids that genes at the head and at the tail are split



Universität Augsburg University

- a random mask is generated, e.g. ABAABABBAB
- the mask determines which bits are copied from which parent

Parents: 1010001110 0011010010

Offspring: 1010000010 0011011110

- *which crossover operator to choose?*
  → trade-off between exploration due to the introduction of new combinations of features and exploitation by keeping the good feature combinations of existing solutions

- generate new offspring from single parents
- *alternatively:* mutate recombined offspring

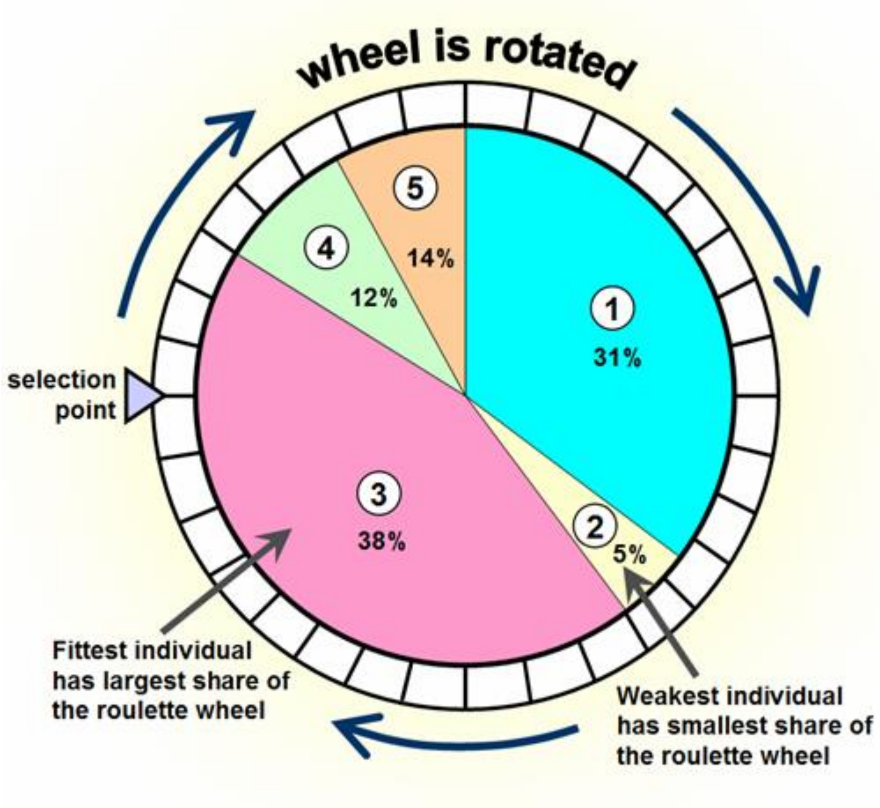
Parent: 1010001110

↓

Offspring: 1011010010

- ensures diversity of the gene pool
  - crossover can only explore combinations of the existing genes
  - mutation can "generate" new genetic information

- *aka:* fitness-proportionate selection
- probability of selection is proportional to individuals' fitness values

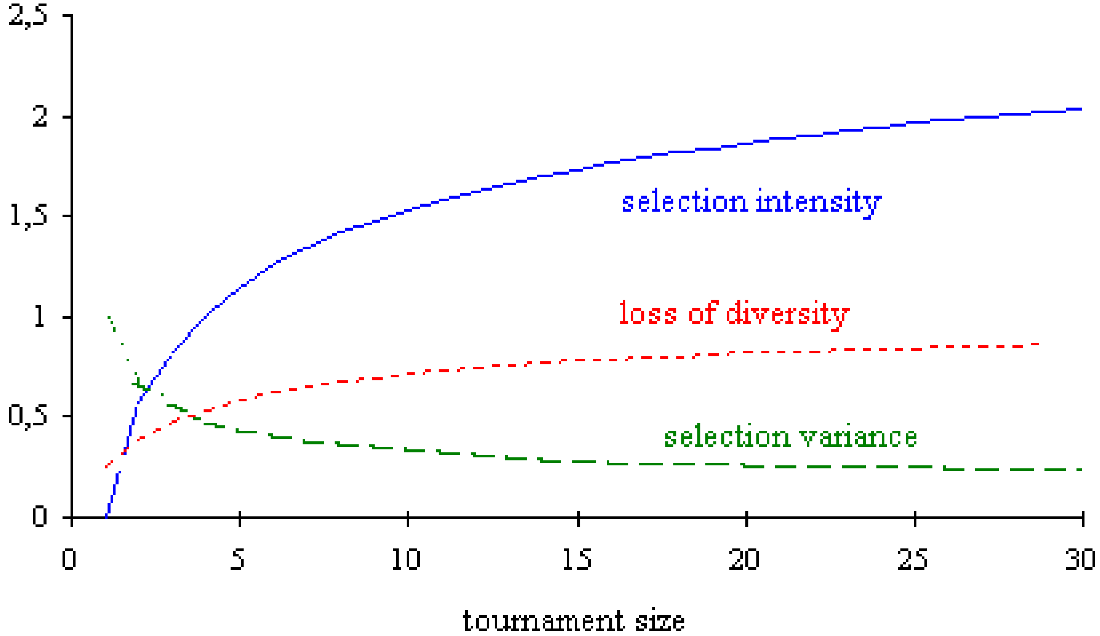


Genetic operators: Rank-based Selection

- rank individuals according to their fitness values
- each rank has a predefined selection probability
  - linear ranking vs. exponential ranking

→ selection based on but not proportional to fitness

- randomly evaluate sets of individuals (2 or more)
- the one with higher fitness is selected as parent
- considers multiple objectives



Pohlheim, H. (2006). GEATbx: Genetic and Evolutionary Algorithm Toolbox for use with MATLAB. Last access on June, 10, 2014.

- *selection intensity or selective pressure:*
  change in average fitness due to selection

- *niche count:*
  the number of points in the population within a certain distance
  → the higher the niche count, the lower the fitness/rank

- *elitism:*
  automatically transfer the best genotype(s) to the next generation

Blickle, T., & Thiele, L. (1995). A comparison of selection schemes used in genetic algorithms.

Discussion: GA parameters

- upfront, as always: problem-specific!

- greater population size:
  - increase in diversity
  - increase in computation time

- greater crossover probability:
  - increase the exploration of the search space
  - increase the chance of disruption of good combinations

- greater mutation probability:
  - helps to introduce new genes or reintroduce lost ones
  - closer to random search