Evolutionary Algorithms

D. Keller and G. Kress

spring semester 2015

Content

Mechanisms of Natural Evolution

P.J. Bentley (1999), *Evolutionary Design by Computers* :

Quote (P.J. Bentley)

> *. . . , as long as some individuals generate copies of themselves which inherit their parents' characteristics with some small variation, and as long as some form of selection preferentially chooses some of the individuals to live and reproduce, evolution will occur.*

Characterization of Natural Evolution

- Evolutionary mechanisms concern a set (*population*) of *individuals*.
- A population is subject to (*Environmental pressure*).
- The information, characterizing an *individual*, is stored compact in its *genotyp*.
- The occurrence of certain properties of an individual (phenotype) is determined by the genotype.
- Phenotype properties decide over the *Fitness* of an individual. The *Fitness* of an individual is a measure of its ability to meet the environmental pressure.

Design Encoding

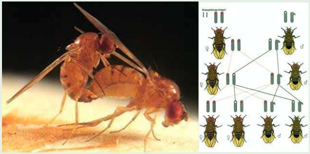
- Nature stores the genetic information by use of a simple alphabet consisting of four basic modules.
- The information does not change throughout an individual's life.
- Die information encodes a *blueprint* of an individual.
- The translation of the code to an individual is called *mapping*.

Example: Genotype

Reproduction) and Inheritance

- Offspring carry genetic information of their parents.
- The information can be transferred by just making copies (cloning) or by *recombination* i.e. (*crossover*). In case of recombination, the offspring carries information of both or, more general, several parents.

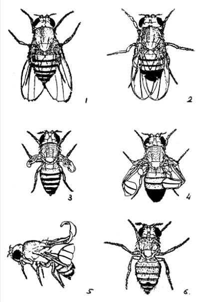
Example: Reproduction



Variation

Some part of the information may be altered bei transferring it from the parent individuals to the offspring (*mutation*).

Example: Variation



Selection

Individuals are selected from the populations on two occasions:

Mating Selection: Which individuals may reproduce?

Environmental Selection: Which individuals survive?

Example: Selection

From natural evolution to an evolutionary algorithm

- Initialization
- Mapping
- Evaluation
- Mating Selection
- Reproduction (Recombination, Mutation)
- Environmental Selection
- Stopping criterion

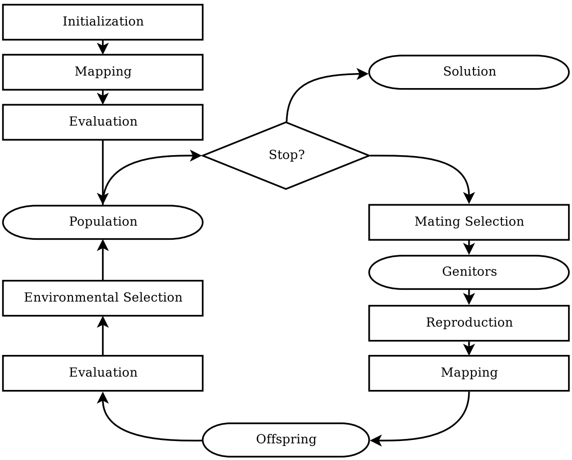
From natural evolution to an evolutionary algorithm

- Initialization
- Mapping
- Evaluation
- Mating Selection
- Reproduction (Recombination, Mutation)
- Environmental Selection
- Stopping criterion

Evolutionary algorithm

```
Generate a starting population {x_i} ∈ 𝕏
Evaluate F(x_i)
Initialize the memory M ← {(x_i, F(x_i))}
Initialize the iterations counter t ← 0
Assign a fitness value F_i to each x_i
while continue(M, t) = 1 do
    M′ ← Mating Selection aus M
    M″ ← Mutation and recombination from M′
    Evaluate F(x̂_i) for all x_i in M″
    Increment t ← t + 1
    M ← Environmental Selection from M and M″
end while
Output the best solution x* in M from
```

Evolutionary Algorithm



The Gene

Definition (Gene)

A gene is an information unit. A gene is not divisible, i.e. the information stored in one gene is transferred as a whole. A gene is equivalent to an optimization parameter $\mathbf{x}_i$ in mathematical programming.

Definition (Allele)

An allele is a state which a gene can take on.

Beispiel (Boolean gene)

A boolean gene $\mathbf{x}_i$ possesses two alleles: **TRUE** und **FALSE**.

Terminology

Individual: A possible solution, i.e. a structure composed of genes.

Population: A set of individuals

Generation: The population M within the evolutionary cycle (Step 11).

Generation number: Iterations counter t

Genotype space: The search space $\mathbb{X}$

Phenotype space: The decision space $\mathbb{D}$

Fitness: The objective- and constraining-functions values $F(\mathbf{x})$, $F(\mathbf{g})$,$F(\mathbf{h})$, often structured like the pseudo-objective after the exterior-point penalty method

Representation

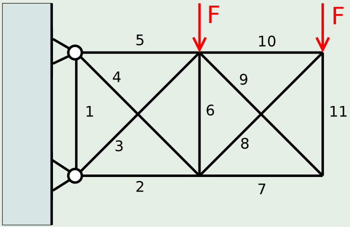
- Representation describes the kind of search space $\mathbb{X}$. Which phenotype properties are encoded with which genes? What kinds of genes are used?
- The Representation corresponds with the parameterization of mathematical programming.

Representation

Example Truss Framework Topology

Beispiel (Truss framework)

Search for a truss framework which is as light as possible without having its maximum displacement exceed the admissible displacement u_{adm}. Available trusses have cross-sectional stiffness $E{\cdot}A = 1$. The node positions are fixed whereas the number of trusses is not. The horizontal and vertical trusses are of length 1.



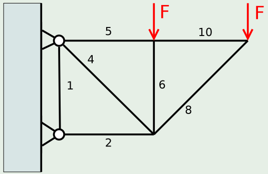
Representation

Example Truss Framework Topology

Beispiel (Truss framework)

The ground structure of the truss framework consists of 11 trusses. A very simple representation of a truss framework consists of a vector with 11 Boolean genes (bit string). Gene *i* determines whether truss *i* exists or not.

1	1	0	1	1	1	0	1	0	1	0



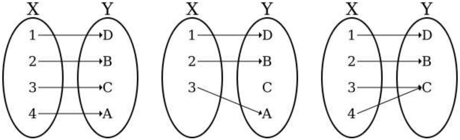
Initialization

- The initial population M must be filled with a defined number of individuals.
- The initial population must include a high genetic diversity (*search space scanning*).
- Genotypes coding illegal or infeasible solutions may be filtered out.
- The initialization can be based random or known solutions.

Mapping

- Mapping describes the translation from the genotype space to phenotype space $\mathbf{y} = f(\mathbf{x})$.
- Ideally, the mapping is bijective. If that can not be reached, remains the requirement of a surjective mapping.

$$\begin{aligned} \text{injective:}\quad & \forall \mathbf{x}, \mathbf{x}' \in \mathbb{X} | f(\mathbf{x}) = f(\mathbf{x}') \Rightarrow \mathbf{x} = \mathbf{x}' \\ \text{surjective:}\quad & \forall \mathbf{y} \in \mathbb{Y} \exists \mathbf{x} \in \mathbb{X} \\ \text{bijective:}\quad & \text{injective and surjective} \end{aligned}$$



(a) bijective (b) injective (c) surjective

Mapping
infeasible and illegal solutions

Definition (Illegal Solution)

A solution which exists in genotype space and which cannot be mapped into pheotype space is called illegal. The number of illegal solutions can be influenced by the kind of representation. Ideally they can be avoided at all.

Definition (Infeasible Solution)

A solution violating one or several constraints is called infeasible. The number of infeasible solutions can be influenced by the kind of representation. Infeasible solutions may, nevertheless, contain valuable information for the development of good and feasible solutions.

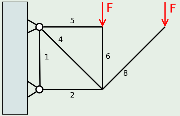
Mapping
Example Truss Framework Topology

Beispiel (Illegal Truss Frameworks)

The introduced representation allows truss frameworks which are not actually load-carrying systems anymore:

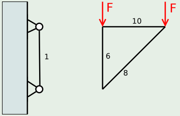
- Mechanisms:

1	1	0	1	1	1	0	1	0	0	0



- Not connected components or loads:

1	0	0	0	0	1	0	1	0	1	0



Beispiel (Infeasible Truss Framworks)

The introduced representation allows infeasible truss frames violating the constraint of an admissible maximum displacement:

$$\max_i(u_i) > u_{adm}$$

Evaluation

- Evaluation is the process of assigning an individual $\mathbf{x}$ a fitness $F(\mathbf{x})$.
- Mostly, the evaluation concerns the objective and constraining functions of one individual. Depending on the problem at hand, the fitness of one individual may also contain information regarding its relation to other individuals of the same generation, for instance to promote genetic diversity or to effect solutions to be evenly-spaced along a Pareto-optimum edge.
- In structural optimization the evaluation step usually implies a structural analysis with FEM, wherefore the evaluation requires most of the computer resources during the optimization process.
- Since always a population of individuals is to be evaluated at one time, the evaluations can easily be partitioned to several computers, or processors.

Evaluation

Example Truss Framework Topology

Beispiel (Truss Framework Evaluation)

- Since illegal configurations may appear, the solution must be checked before the evaluation. Illegal solutions can be equipped with an error fitness. An error fitness is a very high value, making sure that illegal solutions can not establish themselves.
- The evaluation can be performed with FEM; i.e. a truss framework model must be created and solved.
- Mass and maximum displacement determine, for instance through the exterior-point penalty method, a scalar fitness value.

Mating Selection

The mating selection chooses from a population those individuals who will later on reproduce.

Stochastic Selection Each individual i is assigned a probability p_i of being chosen. Then, n individuals are chosen according to their probabilities.

Deterministic Selection Choose the n fittest Individuals.

Mating Selection
Selection Probabilities

Example: Fitness Proportional Selection

- $p_i = 1 - \frac{F_i}{\sum_{j=1}^{size(M)} F_j}$
- Disadvantage: Sensitive to adding a constant to the fitness.

Example: Rank-based Selection

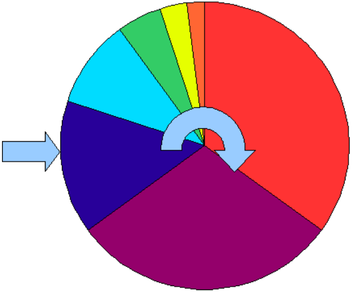
$$\begin{aligned}\hat{p}_i &= \alpha\,(size(M) - \mathrm{rank}(F_i)) + \beta \\ p_i &= \frac{\hat{p}_j}{\sum_{j=1}^{size(M)} \hat{p}_j}\end{aligned}$$

- Invariant with respect to the difference between fitness values.

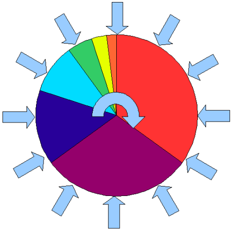
1. T individuals are chosen randomly. T is called a tournament size.
2. Of those T individuals the fittest one is copied into the population M'.
3. The process is repeated until M' has reached size n.

Mating Selection
Roulette-Wheel Selection

- The probabilities p_i are proportionally partitioned on a roulette wheel.
- As many drawings as individuals are to be chosen are performed.



- For choosing n individuals, the roulette-wheel selection requires n drawings and n random numbers. This leads to a large variance of the selection results if the process is repeatedly applied to the same parent generation.
- Stochastic Universal Sampling chooses in one drawing n individuals so that a smaller variance is to be expected.



Reproduction

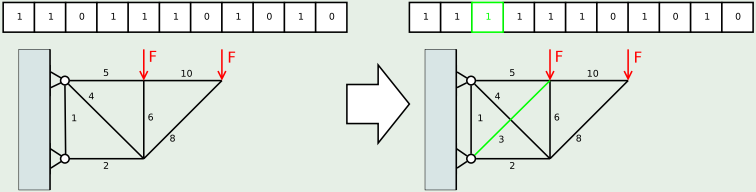
- Reproduction generates new solutions (Offspring) by variation of the selected parent individuals.
- There are two main categories of variation:

  Mutation: one parent individual

  Cross-over: two (or more) parent individuals

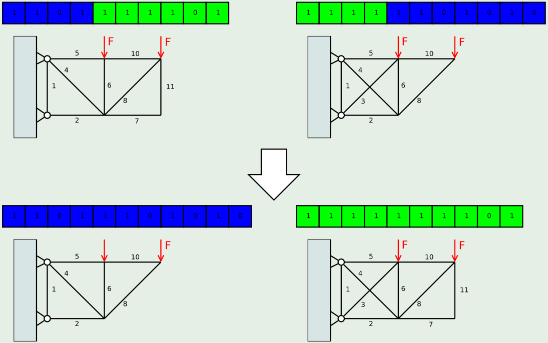
Beispiel (Mutation at the Truss Framework)

Randomly a parent gene *i* is determined whose value is altered.



Beispiel (One-Point Crossover at the Truss Framework)

Randomly a parent gene i is determined. Genes 1 through i of the first parent individual recombine with genes $i+1$ through 11 of the second parent individual and vice versa.



Environmental Selection
Replacement

The environmental selection chooses from the offspring as well as from the parent population those individuals who may be talen to the next generations, or survive.

- The same mechanism as used for the mating selection are applicable.
- Often a deterministic selection is applied:
  - Comma: The best individuals of the offspring form the new generation.
  - Plus: The best individuals of the offspring and the parent generation as a whole form the new generation.

Termination

There is no applicable sufficient criterion for identifying a global optimum. Thus, an artificial criterion is applied either by itself or in combination with others:

- termination after t_{max} generations
- termination after k_{max} function evaluations
- termination after h time units.
- termination after the fitness of the best individual has crossed a predefined value F_{ult}.
- termination if the fitness did not further improve within the last t_s generations

Advantages of Evolutionary Algorithms

- No assumptions on the objective function necessary (Black-Box)
- Acceptable performance for a large selection of problem types
- Universally applicable
- Robust with respect to illegal solutions through parallel computing
- Little development effort, easy to parallelize
- Issues a set of solutions
- Well suited for multi-objective optimization (spanning of the Pareto edge)

Limitations of Evolutionary Algorithms

- There is no guarantee that a global optimum is found in a finite time
- Weak theoretical basis (Heuristic)
- Often a large number of parameters for influencing convergence
- Often a large number of evaluations necessary to find good solutions

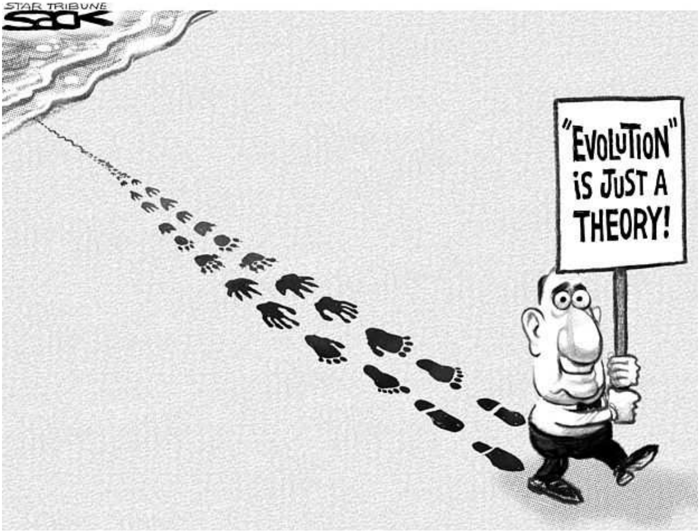
Application Fields of Evolutionary Algorithms

- Numerical and combinatorial optimization
- System modeling
- Control
- Structural optimization
- Software development
- Adaptive machines and programs
- Artificial intelligence

Research Topics

- Multi-objective optimization
- Hybridization: Embedding of local search methods (i.e. mathematical programming)
- Embedding of adaptive mechanisms: Learning algorithms
- Parallel and distributed evolutionary algorithms (island systems)
- Uncertainties of evaluation und robustness assessment
- Application to economical and social simulations
- Abstract representation concepts

Questions?



Content

- Name the key components of an evolutionary algorithm as inspired by biology.
- What is an *individual*?
- What is a *population*?
- What is meant with *environmental pressure*?
- What is a *Genotype*?
- What is a *Phenotype*?
- What is meant with *fitness*?
- What are *selection*/*reproduction*/*inheritance*/*variation*?
- What is meant with *evaluation*?
- Describe an evolutionary algorithm.
- What is an *gene*/*allele*?
- What is a *generation*?
- What is meant with *representation*?
- What must be kept in mind when initializing a starting population?

- What is meant with *mapping*? Which properties desires one of the *mapping*?
- What is an *illegal solution*?
- What is an *infeasible solution*?
- What is meant with *mating selection*?
- Describe stochastic and deterministic selection.
- Name two kinds of defining the selection probabilities for one single individual in stochastic selection.
- How does the tournament selection work?
- How does the roulette-wheel selection work?
- What is *stochastic universal sampling*?
- Name the two most important variation strategies.
- What are *mutation*/*crossover*?
- What is *environmental selection*?
- Name an example for a termination criterion for an evolutionary algorithm.
- Discuss Advantages and limitations of evolutionary Algorithms with respect to mathematical programming.