Genetic Algorithms Following Selection, Mating and Mutation

Alexander Schatten

January 14, 2010

Institut für Softwaretechnik & Interaktive Systeme

Selbstorganisierende Systeme WS09/10

Folien übertragen[1] in Latex / Lyx. - modifiziert/gekürzt und möglicherweise mit Fehlern

Contents

[1] von Georg Regal

1 About Optimisation & Learning

- Learning as Optimisation
- Modeling of Natural Processes
- Topographical Illustration
- Optimisation-Methods
  - Deterministic
  - Random
  - Brute Force
  - Evolutionary

1.1 What is Optimisation?

- Maximize Yield
- Minimize Effort
  - Travelling Salesman
- Calculate optimal parameters of mathematical models
  - Time-series analysis
  - Pattern matching
- Finding models
  - Curve fitting
  - Traffic routing
  - Pattern matching
- Categories of Optimisation Problems
- Parameter Estimation
  - Mathematical models
- Subset Selection
  - Select a subset of a larger set
- Combinatorial problems (Sequencing problems)
  - Travelling Salesman

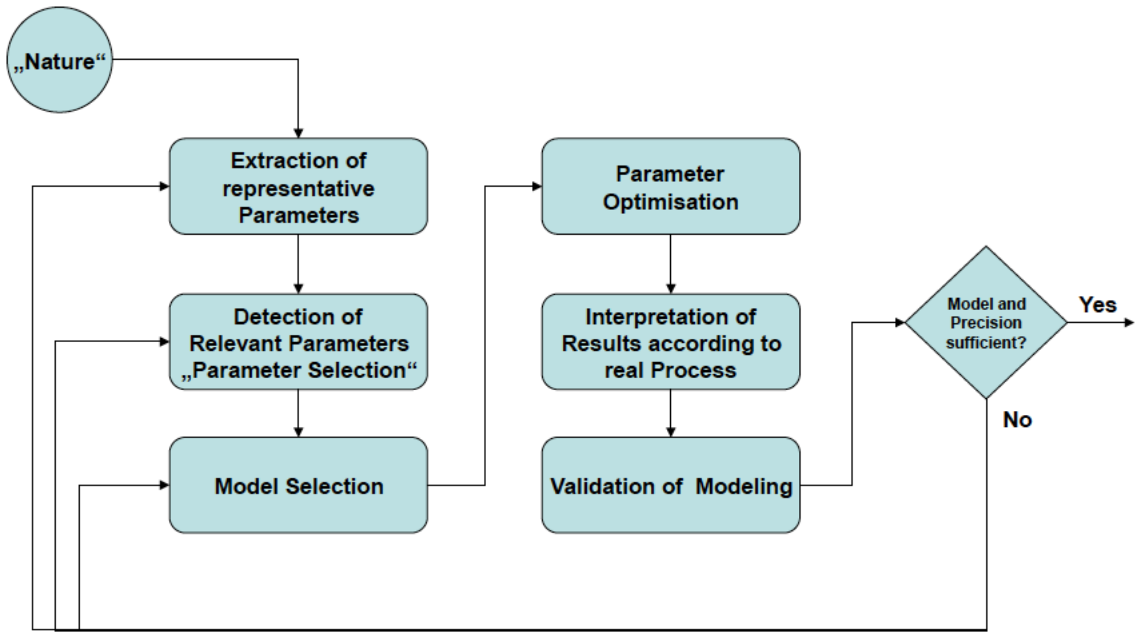

Figure 1: Modeling

1.2 Quality Function and Phase Space

- A Quality Function is a function, that allows to assigne a quantitative quality measure to a specifec set of parameters or a specific configuration or status of a system
- Phase Space is the n-dimensional space, that is spanned by the relevant parameters of a system - Fig.2P.4

1.3 A Variation of Optimisation Methods

- Full Grid Search (Brute Force)
- Hill Climbing (deterministic)
  - Gauß-Seidel
  - Simplex
  - Gradient Method
- Radom Search (non-deterministic)
  - Monte-Carlo Methods
- Combinations
  - Evolutionary Algorithms
  - Genetic Algorithms
  - Simmulated Annealing ?

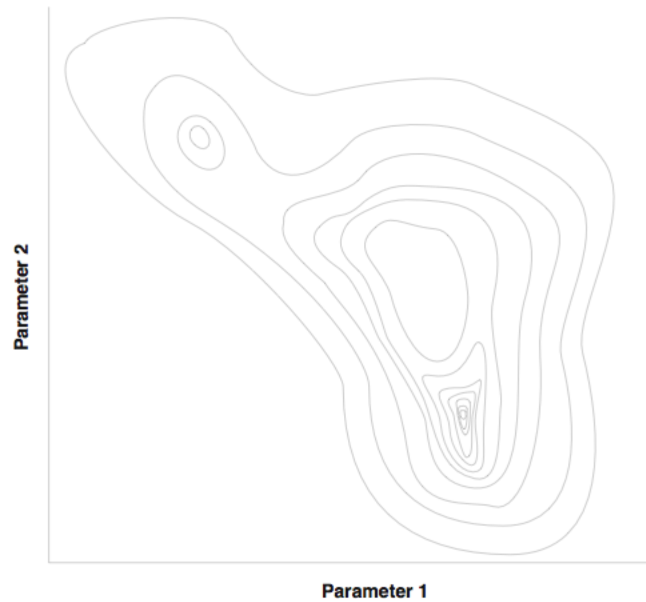

Figure 2: Topography

1.4 Biological and Mathematical Mechanisms

Biology	Mathematical Approach
Selection	Hillclimbing
Mating / Crossover	Random
Mutation	Search

Genetic Principles

- Genotype –> Phenotype
- Phenotype is determined by the information coded in Gens (plus the environment during ontogenesis)
- Mitose/Meiose
  - Sexual reproduction vs. asexual reproduction
  - Crossover

2 Genetic Algorithms

- Coding (example: parameter estimation)
- Algorithm principles
  - Selection
  - Mating
  - Crossover
  - Mutation

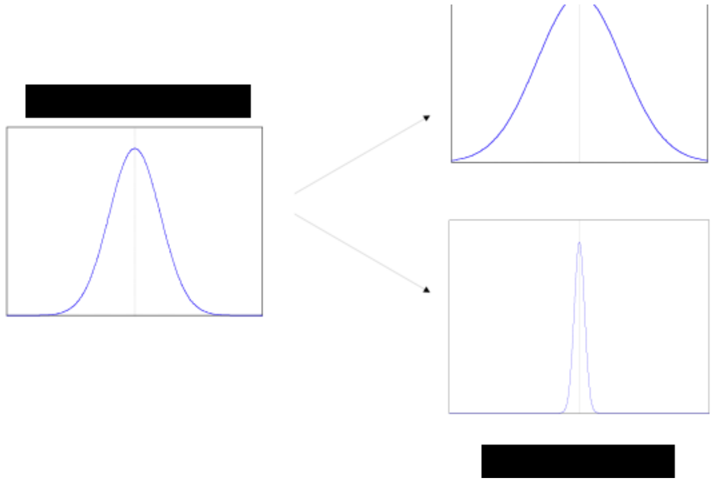

Figure 3: Phenotype –> Mutation (oben) bzw. Selection (unten)

2.1 Binary Bitstrings and Graycode

Parameter 1 0,07
Parameter 2 0,01
Parameter 3 0,1
Parameter 4 0,13

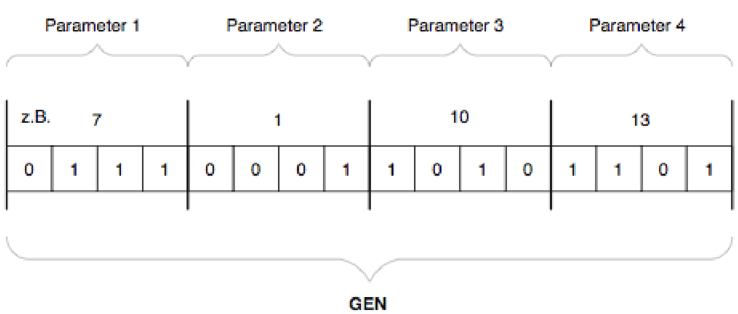

Figure 4: GEN

Binary	Gray	Decimal
000	000	0
001	001	1
010	011	2
011	010	3
100	110	4
101	111	5
110	101	6
111	100	7

Figure 5: Graycode

2.2 Algorithm

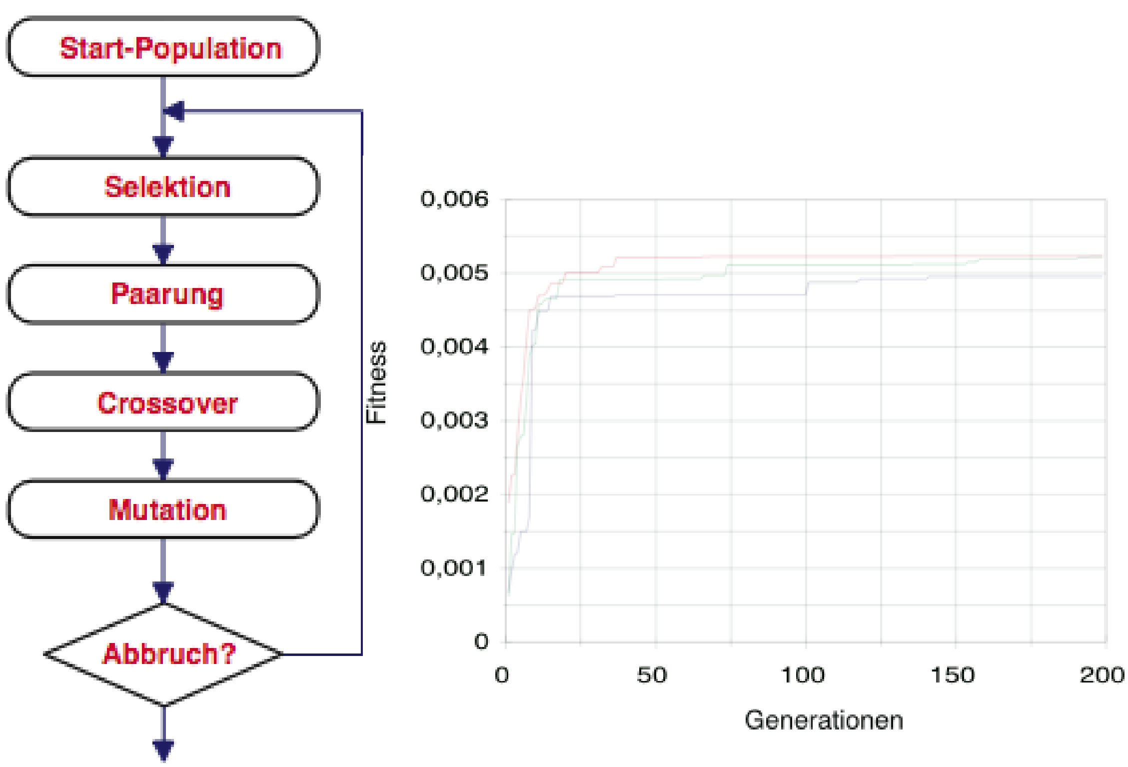

Figure 6: Algorithm

2.3 Crossover

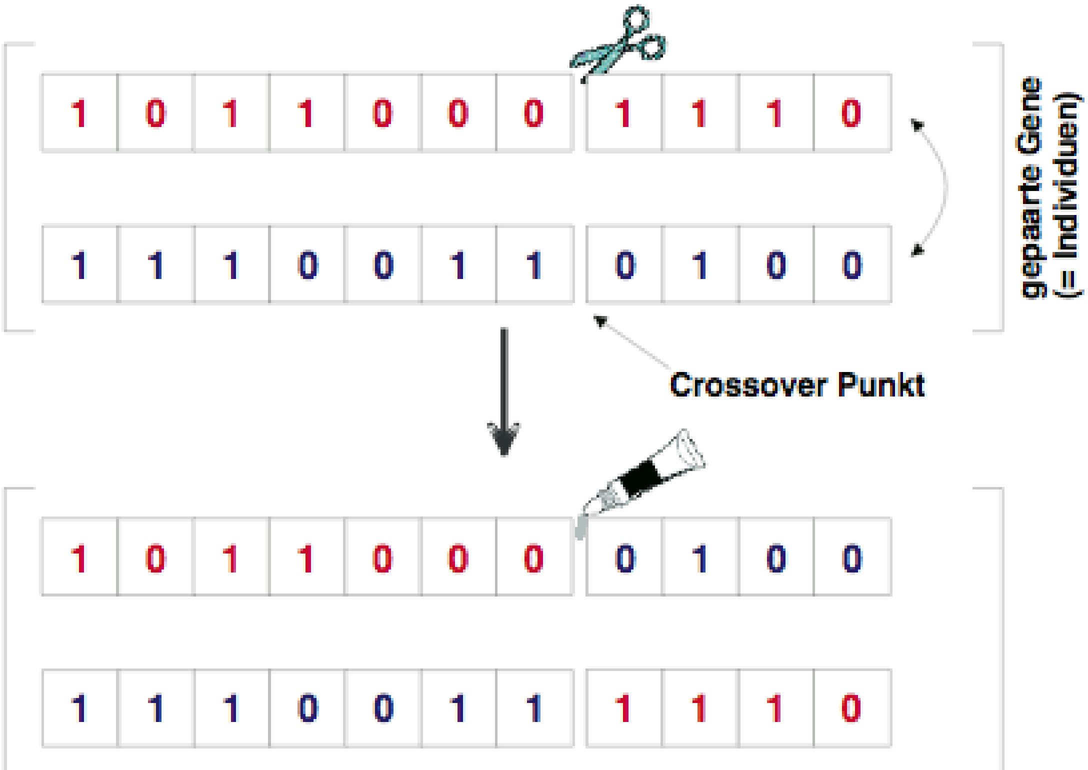

Figure 7: Crossover

One Point C
Two Point C
Uniform C

3 Evolutionary Algorithms

3.1 General Ideas

- Do not „stick“ too much on a specific „school“, „concept“...
- Ideas are:
  - Adaption
  - Bring new „ideas“ into a system (Mutation)
  - Select among competitors (Selection)
  - Bring together successful individuals (Mating)

3.2 German vs. American „School“ E.g. Ingo Rechenberg (Germany) vs. Goldberg (U.S.A)

- Coding!
  - GA: Bitstrings

- EA: Vector

- EA Algorithm

  - (1+1) - EA
  - (μ+λ) - EA
    - μ parents create λ offsprings
    - Fittest survive
    - (potential „eternal“ life)
  - (μ,λ) - EA
    - Like above, selection only from offspring
    - „Death“ part of the model

3.3 Parallel Algorithms

- GAs are implicite parallel

- Multitasking/-threading

- Distributed Computing

- Multi-Population Models

  - Island
  - Network
  - Migration
  - ....