UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft

Genetic Algorithms

Annette Bösherz, Henning Gallicki, Christian Kipp, Harry Kran,
Katharina Miluchin, Sebastian Schweer

Agenda

1. Motivation
2. Simple Genetic-Algorithm
3. Recombination
4. RAR-Mutation
5. VRP-Solution
6. Conclusion

Motivation

- Proposed by J. Holland [1975]
- Applicable to ill-defined problems
- Imitates *natural selection*:
  - Chromosom = particular solution
  - Evolution = Exclusion of worst solutions

Simple Genetic-Algorithm

Initial: $X^1 = \{x_1^1, \dots, x_N^1\}$

for t = 1 to T

for j = 1 to k

Step1: **Reproduction** (Selecting two parent chromosomes from X^t)

Step2: **Recombination** (Generate two offspring form the two parents chromosomes using a crossover operator.)

Step3: **Mutation** (Apply a random mutation to each offspring with small probability)

Step4: **Generation Replacement**

(removing the worst solution in X^t and replace by offspring.)

Recombination

One-point crossover

Parent 1 :	0	1	2	3	4	5
Parent 2 :	0	4	3	2	5	1
Offspring 1:	0	1	2	2	5	1
Offspring 2:	0	4	3	3	4	5

Order crossover(OX-Method)

Parent 1 :	0	1	2	3	4	5
Parent 2 :	0	4	3	2	5	1
Offspring 1:	-	-	-	3	4	-
Offspring 1:	0	2	5	3	4	1

OX-Crossover

Step 1: Select two random Parents

Step 2: Select a String between cut-points form Parent1 and copy it to offspring

Step 3: Filtering all variables existing in the cut String out of Parent2

Step 4: Fill the rest of the offspring with the filtered variables of Parent 2

```
l ← 1
m ← 1
► Step 1
create random i
create random j
select parent P1
select Parent P2
if ( i = j ) restart
if ( i > j )   i ←→ j
```

```
For k ← i to j   ► Step 2
     Off[k] ← P1[k]
For k ← 1 to length P2  ► Step 3
     if P2[k] is not Element of  Off [ ]
     X[ l ] ← P2[k]
     l ← l+1
For K← 1 to i-1  ► Step 4
     Off[k] ← X[m]
      m ← m+1
For K ← j+1 to length P2
      Off[k] ← X[m]
      m ← m+1
```

RAR-Mutation

Pseudocode:

```
create random i; create random j
if (i = j) Then restart
if (i < j) Then
    temp ←A[i]
    For k ←i to j-1
            A[K] ←A[K+1]
    A[j] ←temp
Elseif (i > j) Then [analog...]
```

Offspring 1:	0	1	**2**	3	4	5
Offspring 1:	0	1	3	4	**2**	5

What is the VRP ?

• Everybody knows the Traveling Salesman Problem(TSP)

•A Expansion of the TSP, is the mTSP (multiple TSP), where the Cities witch must be visit, are divided to many Travelers.

•The Vehicle Routing Problem(VRP) is a mTSP with additive Capacities.

(A specialization for the VRP is the capacitated VRP (CVRP), where all Transporters have the same Capacity.)

VRP-Solution

- Given a solution to VRP with multiple routes contains multiple copies of the depot, with each copy acting as a separator between two routes
- Example: The String would correspond to a VRP Solution.
- The first route contains vertices 1 and 2, the second route contain vertice 4 and the last route only contains vertex 5.
- A classical crossover operator and a RAR mutation operator are then adapted for this representation.
- Working until the required number of feasible offspring is produced (infeasible offspring are discarded)
- **Representation**:

String :	0	1	2	0	4	0	5

Conclusion

- Solutions of GA, Simulated Annealing and Tabu-Search are of comparable quality
- GA was more computationally expensive than other methods like simple construction heuristics with improvement procedures
- GA are not yet competitive on VRP but on some other similar problems

➔ Further research on VRP could lead to competitive implementations.