

## Sequentielle quadratische Approximation Evolutionäre Algorithmen

# Das Problem der Optimierung

$$\min f(\mathbf{x}) \quad \mathbf{x} \in \mathbf{R}^n$$

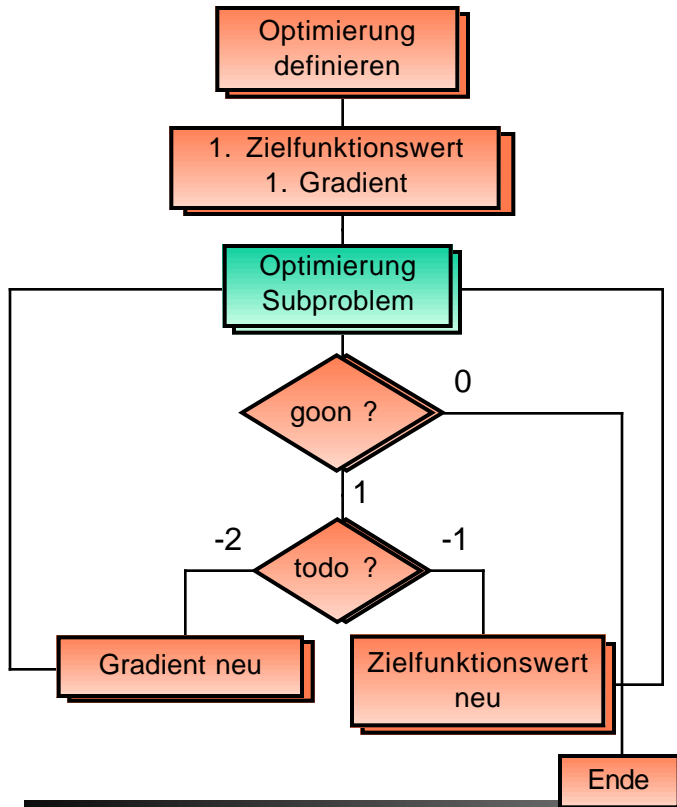
$$h_i(\mathbf{x}) = 0 \quad i = 1, \dots, m_{\text{equal}}$$

$$g_j(\mathbf{x}) \geq 0 \quad j = 1, \dots, m_{\text{inequa}}$$



# NLPQL

## nonlinear programming quadratic linesearch



```
optimize allocate, ... optimize create, ...  
optimize modify, design/bounds/accuracy/  
diff-interval/num-iter
```

```
control gosub, , FUNCTION, /  
control gosub, , GRADIENTS, /
```

```
#label MAIN-LOOP  
optimize search, reverse, , /  
optimize extract, iter-info, , todo goon /
```

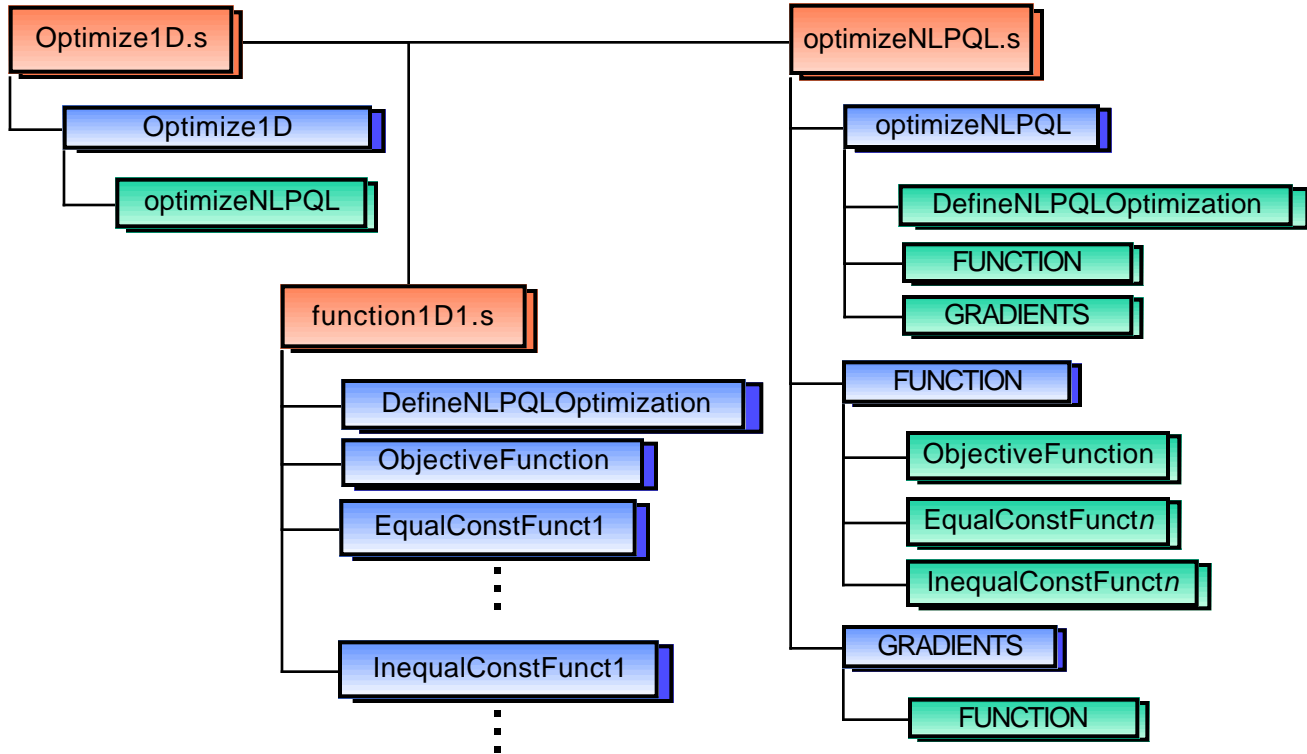
```
control if, equal, goon 0 END, /
```

```
control if, gosub equal, todo -1 FUNCTION, /  
control if, gosub equal, todo -1 GRADIENTS, /  
control if, equal, 0 0 MAIN-LOOP, /
```

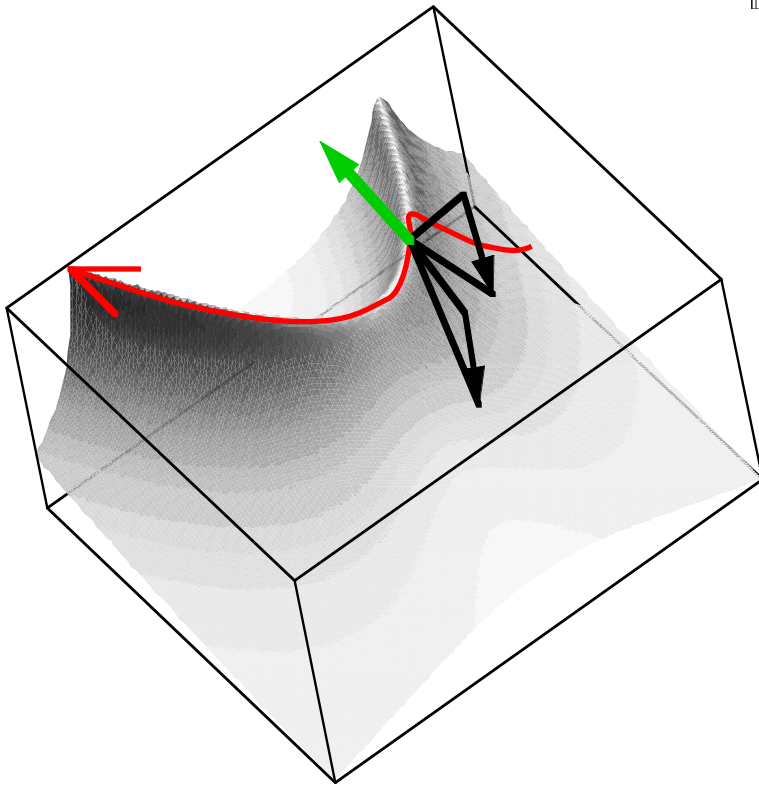
```
#label END-MAIN-LOOP
```



# Dateistruktur Beispiel Optimize1D



# Testfunktion



Auf dem Grat ist der Gradient  
numerisch nicht bestimmbar.

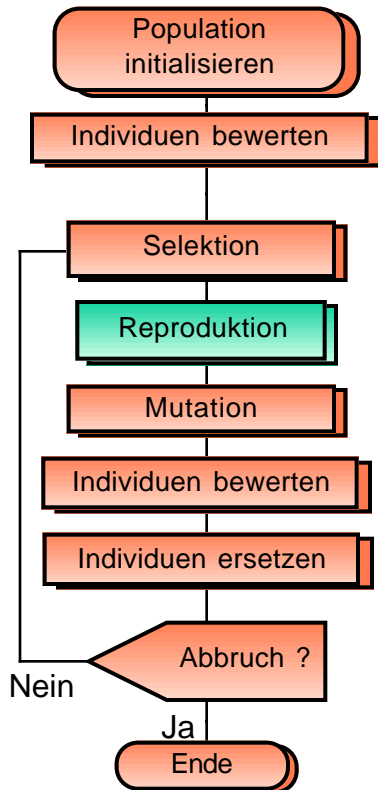
$$z(x,y) = |x^2 + y| + 0.1x$$

Lösungspfad

Anstieg in x- bzw.  
y-Richtung

Gradient

# Evolutions Algorithmus



`genetic allocate, ... genetic initialize, ...`

`control gosub, , evaluateAllNewborns, /  
genetic modify, fitness ...  
genetic next_generation, ...`

`genetic select, ... genetic roulett, ...  
(genetic modify, elit ...)`

`genetic xover, ...`

`genetic mutate, ...`

`control gosub, , evaluateAllNewborns, /  
genetic modify, fitness ...`

`genetic modify, kill ...`

## Weitere Kommandos

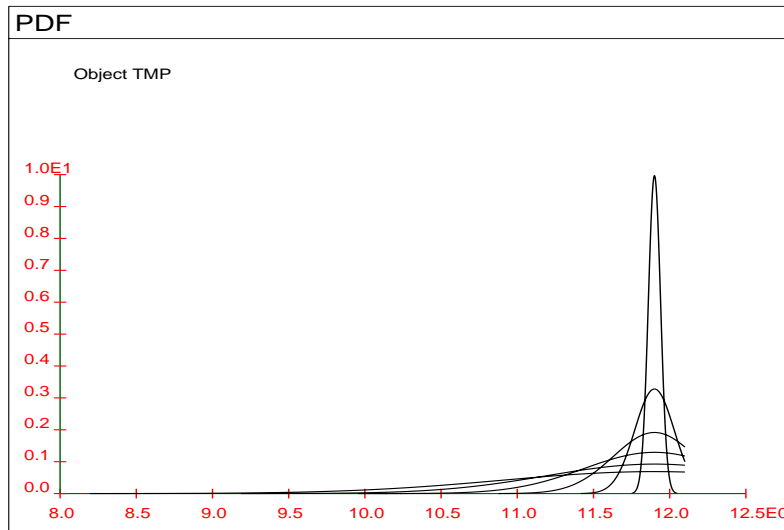
`genetic statistics, ...  
genetic read, ...`



# Mutation

gewünschte Eigenschaften der Zufallsvariable für die Mutation:

- nicht gleichverteilt
- Mutationsschrittweite anpaßbar
- gültiger Bereich



Slang - the Structural Language, Version 3.5. Created on 24-Oct-1998.



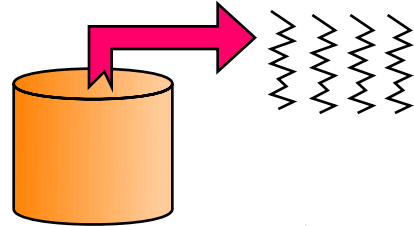
# Mutation truncated\_normal

```
ranvar allocate, replace, 2, /  
ranvar create, truncated_normal, 1 0 1 -1 1, rvX /  
ranvar create, truncated_normal, 2 0 1 -1 1, rvY /  
  
group allocate, replace, 1, /  
group create, ranvars, 1 2 rvX rvY, MutationGroup /  
  
ranset allocate, replace, 1, /  
ranset create, , 1 MutationGroup, MutationSet /  
  
genetic mutate, , 1 .2 MutationSet stddev_vec, /
```

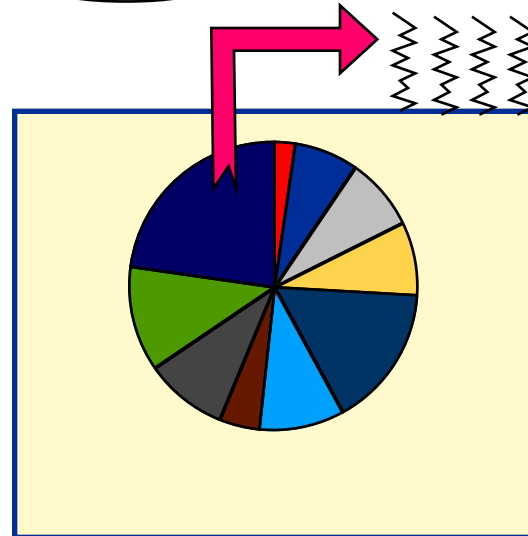


# Selektion deterministisch/stochastisch

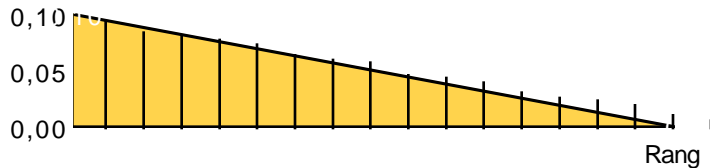
```
genetic select, sel_num low_rank, ... /  
genetic select, sel_num nosort, ... /  
ziehen aus einer Urne ohne zurücklegen
```



```
genetic roulette, sel_num nopressure, ...  
ziehen aus einer Urne mit zurücklegen
```



```
genetic roulette, sel_num rank linear, ...
```



Rang

Zeile 1

0,1

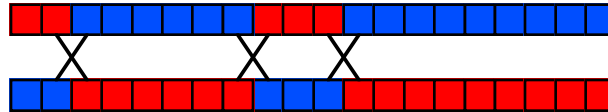
Rang

0

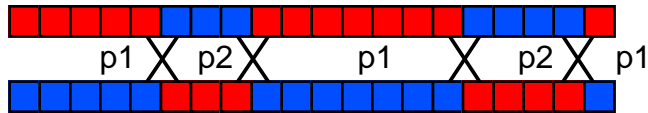


# Crossover Verfahren

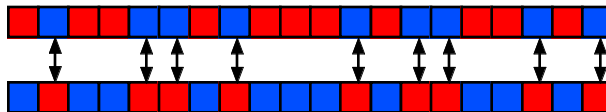
genetic xover, **multipoint**, 1 parents 3, /



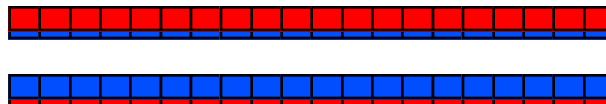
genetic xover, **segmented**, 1 parents p1 p2, /



genetic xover, **uniform**, 1 parents .5, /



genetic xover, **arithmetic**, 1 parents .3, /



# Straffunktion

$$\bar{f}(\mathbf{x}) = \begin{cases} f(\mathbf{x}) & \text{wenn } g(\mathbf{x}) \geq 0 \\ 0.5 & \text{wenn } g(\mathbf{x}) < 0 \end{cases}$$