Vortrag über

# Genetische und Evolutionäre Algorithmen (Vol. 2)

von

Adam El Sayed Auf und Kai Lienemann

**Gliederung:**

4) Rückblick

5) Allgemeine Einführung

6) Genauere Beschreibung von Evolutionären Strategien

7) Vergleich der beiden Techniken

8) TSP – Beispiel

9) Memetische Algorithmen

10) Animationen

# Rückblick

- Schritt 1: Initialisierung
- Schritt 2: Ausgangslösung bewerten
- Schritt 3: Selektion und Replikation
- Schritt 4: Erzeugung von Nachkommen durch:
  - » Partnerwahl
  - » Crossover
  - » Mutation
  - » Bewerten und Hinzufügen
- Schritt 5: Schritt 3 bis zum Abbruchkriterium

# Einführung

- **Die 60er Jahre**
- **Ingo Rechenberg**
- **Hans-Paul Schwefel**
- **Zwei Häuser…**
- **Schwefel und ein paar neutrale Theoretiker**

# Rechenbergsche Grafik-Notation

- Individuensymbol
- Population
- Gleichverteilte Auswahl
- Duplikation
- Auswahl nach Qualifikationsfunktion
- Mutation
- Rekombination und Mischung
- Phänotypische Realisierung
- Bewertung
- Isolierte Population

# Codierung

- an Realzahlen orientiert
- Vektoren von reellen Zahlen
- Individuum entspricht einem solchen Vektor
- Eine Population entspricht einer Menge von Vektoren
- Rechenberg der Ingenieur
- kompakteste Form der Codierung

# Die ($\mu$+$\lambda$)-ES

- Verallgemeinerung der (1+1)-ES
- $\mu$ Eltern
- $\lambda$ Kinder
- Bedingung: $\lambda \geq \mu \geq 1$
- Auswahlwahrscheinlichkeit ist gleich
- konstante Population
- die Qualität der Besten wird immer besser
- Eltern erzeugen $\lambda$ Mutanten; die $\mu$ besten überleben

# Die (μ,λ)-ES

(Schwefels Dissertation 1974)

- Aufgabe der Unsterblichkeit
- mehr Realitätsnähe
- trotzdem können gravierende Nachteile entstehen
- monotone und nicht monotone Qualitätsfunktionen
- Beeinflussung durch geringfügige Mutation
- Ist die ES nicht von Bedeutung: (μ#λ) – ES
- # entspricht hier „+“ oder „ , “

# Selektionsdruck

- $s = (\mu/\lambda)$ bestimmt den Selektionsdruck
- je kleiner der Quotient, desto stärker ist der Selektionsdruck
- je näher der Quotient an 1 liegt, desto kleiner der Druck
- somit ist eine Selektionsdruckeinstellung möglich
- Populationswellen
- Populationswellen mit schwankendem Selektionsdruck
- zyklische Variation des Quotienten s

# Die ($\mu/\rho\#\lambda$)-ES

- sexuelle Rekombination
- $\mu$ Eltern erzeugen $\lambda$ Kinder
- Gruppen aus der Elternpopulation zeugen Nachkommen
- $\rho$ entspricht der Anzahl der Gruppenelemente (z.B. $P = 2$)
- zwei unterschiedliche Rekombinationsstrategien
- Mittelwerte und diskrete Vertauschung

# Populationen

- Erweiterung des Formalismus auf Populationen
- [2,3(4,7)]-ES
- Populationsqualität
- „no limits“ [4/2#3(5#7)]-ES (Genfluss)
- isolierte Populationen: $[1,2(4,7)^{/30}]$-ES
- Paralleles durchsuchen des Suchraums

# Mutation

- Gleichverteilung und Normalverteilung
- Erzeugerselektion nach Gleichverteilung
- Mutation nach statistischer Normalverteilung
- Mutative Schrittweitensteuerung
- $x_{neu} = x_{alt} + N(0,\sigma)$ „Erzeugung mutierter Nachkommen“
- $N(0,\sigma)$ Vektor Gaußverteilter Zufallsvariablen
- $\sigma$ ist der Streuungsparameter
- $x_{alt} = \langle 0.3, -0.6, 2.5 \rangle$ $\sigma = \langle 1, 0.5, 6 \rangle$
- $x_{neu} = \langle -0.8, 0.1, -3.2 \rangle$

# Vergleich

| **ES** | **GenA** |
|---|---|
| Schnelle lokale Optimierung | Durchforstung des Suchraums |
| Erreicht schnell homogene Population | Vermeidet genau diesen Effekt |
| Starke Mutation | Vermeiden von Mutation |
| Starke Selektion | Schwache Selektion |

# Anwendung des GA beim TSP

- Vorraussetzung:
  - Lösungsrepräsentation
  - Crossoveroperator
  - Mutationsoperator

# Lösungsrepräsentation

**Erster Ansatz:**

- Codierung als binären Vektor
- Weg als Kanten zwischen den Städten A-F

| ab | ac | ad | ae | af | bc | bd | be | bf | cd | ce | cf | de | df | ef |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 |

- formal ungültige Lösungen

| ab | ac | ad | ae | af | bc | bd | be | bf | cd | ce | cf | de | df | ef |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

# Lösungsrepräsentation

**Zweiter Ansatz:**

- Darstellung des Weges als ein Vektor von Indizes
- Jeder Stadt wir ein Index zugeordnet
- Bei k Städten muss der Vektor alle Indizes von 0 bis k-1 mal enthalten und nur genau einmal

**Beispiel:** $L_1 = ( 4 , 6 , 7 , 1 , 3 , 9 , 8 , 5 , 2 , 0 )$

$L_2 = ( 0 , 1 , 2 , 3 , 4 , 5 , 6 , 7 , 8 , 9 )$

$L_3 = ( 6 , 1 , 9 , 5 , 8 , 2 , 4 , 7 , 0 , 3 )$

# Order Crossover

- Einteilung des Vektors in drei Intervalle
- Austausch der Informationen im mittleren Teil
- Wiederhole für jeden doppelten Wert w in $A_1$:
  - Suche Position p von w in B
  - doppelter Wert ausserhalb des Intervalls wird durch den Wert ersetzt, der in A an der Position p steht

# Order Crossover angewandt

$$\mathbf{A} = \begin{array}{ccc|ccc|cccc} 8 & 7 & 3 & 4 & 5 & 6 & 0 & 2 & 1 & 9 \end{array}$$
$$\mathbf{B} = \begin{array}{ccc|ccc|cccc} 7 & 6 & 0 & 1 & 2 & 9 & 8 & 4 & 3 & 5 \end{array}$$

$$\mathbf{A}_1 = \begin{array}{ccc|ccc|cccc} 8 & 7 & 3 & 1 & 2 & 9 & 0 & 2 & 1 & 9 \end{array}$$
$$\mathbf{B}_1 = \begin{array}{ccc|ccc|cccc} 7 & 6 & 0 & 4 & 5 & 6 & 8 & 4 & 3 & 5 \end{array}$$

$$\mathbf{A}_1 = \begin{array}{ccc|ccc|cccc} 8 & 7 & 3 & 1 & 2 & 9 & 0 & 5 & 4 & 6 \end{array}$$
$$\mathbf{B}_1 = \begin{array}{ccc|ccc|cccc} 7 & 9 & 0 & 4 & 5 & 6 & 8 & 1 & 3 & 2 \end{array}$$

# Cycle Crossover

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| **A** = | 8 | 7 | 1 | 0 | 6 | 3 | 4 | 9 | 5 | 2 |
| **B** = | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| **A** = | 8 | - | - | 0 | - | 3 | - | - | 5 | - |
| **B** = | 0 | - | - | 3 | - | 5 | - | - | 8 | - |

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| **A** = | 8 | 1 | 2 | 0 | 4 | 3 | 6 | 7 | 5 | 9 |
| **B** = | 0 | 7 | 1 | 3 | 6 | 5 | 4 | 9 | 8 | 2 |

# Mutationsoperator

- Wähle zwei zufällige Positionen p1 und p2
- lösche die vordere Position p1(=3)

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| **A** = | 8 | 7 | 1 | 0 | 6 | 3 | 4 | 9 | 5 | 2 |
| $\mathbf{A}_1$ = | 8 | 7 | - | 0 | 6 | 3 | 4 | 9 | 5 | 2 |

- Verschiebe die folgenden Werte bis p2(=9) um eine Stelle nach links

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| $\mathbf{A}_2$ = | 8 | 7 | 0 | 6 | 3 | 4 | 9 | 5 | - | 2 |

- Füge den gelöschten Index in die neue Lücke ein

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| $\mathbf{A}_3$ = | 8 | 7 | 0 | 6 | 3 | 4 | 9 | 5 | 5 | 2 |

# Anwendung des GA beim TSP

- TSP für 25 Deutsche Städte
- Bewertungsfunktion verwendet die Distanzen zwischen den Städten
- Beste Wegstrecke von 3294 km nach 137 Generationen erhalten

# Memetische Algorithmen

- Optimierung des genetischen Codes durch Überleben der besten Individuen
- Beispiel für Optimierung ist die Kung-Fu-Kampfkunst
- Genetische Algorithmen inspiriert von genetischer Evolution
- Memetische Algorithmen inspiriert von kultureller Evolution

# Herangehensweise

- Nachahmung der kulturellen Evolution
- Kombination aus:
    - globaler Suche durch eine ganze Population
    - lokale Suche durch jedes einzelne Individuum

# Grundverfahren

- Erzeugen einer zufälligen Population
- Jedes Individuum führt lokale Suche durch
- können sich in einem lokalen Optimum befinden
- Interaktion mit den anderen Individuen:
  - Kooperation (Crossover => Austausch von Informationen)
  - Wettkampf (Selektion)
- Wiederholung bis ein Abbruchkriterium erreicht ist

# Competitve-Cooperative Approach

- Suche in grossen Lösungsräumen
- Eigenschaften unterschiedlicher getrennt suchender Individuen verwenden
- lokale Suche durch Monte Carlo simulated annealing *(Vortrag: Christoph)*
- Kooperation durch Crossover (GA)
- Wettkampf durch Fitness der einzelnen Individuen

# Wettkampf

- Individuen greifen an und werden angegriffen
- Bei Sieg wird der „Verlierer“ aus der Liste entfernt und der „Sieger“ an diese Stelle kopiert

$$p(\Delta E_{comp}, T) = \frac{1}{1 + e^{\Delta E_{comp}/T}}$$

- Zufällige Bestimmung einer Zahl q zwischen 0 und 1
- wenn $p(\Delta E_{comp}(x,y),T) > q$, wird die Tour x gelöscht und mit einer Kopie von y ersetzt

# Kooperation

- Basierend auf dem Crossover bei Genetischen Algorithmen
- Komponenten der Individuen werden ausgetauscht um neues Individuum zu schaffen
- Kann große Änderungen der Fitness der Individuen bewirken
- Spielt eine größere Rolle als die Mutation