

Softcomputing – Biologische Prinzipien in der Informatik

Dipl. Math. Maria Oelinger, Universität Duisburg-Essen
Dipl. Inform. Gabriele Vierhuff, Universität Bremen

www.collide.info/~oelinger



IF TIF 08 – 2003
Evolutionäre Algorithmen

Vorschau

 Geschichte und Einführung

 Das biologische Prinzip

 Grundlagen

 Verschiedene Algorithmen

 Evolutionäre Algorithmen und Neuronale Netze

Geschichte

- 👁👁 Darwin 1859 (Mutation und Selektion)
- 👁👁 Mendel 1865 (Rekombination)
- 👁👁 Wurzeln in der Bionik (Biologie + Technik)
- 👁👁 Nachahmung von natürlichen Vorgängen
 - ⇒ Praktische Verfahren entwickeln
 - ⇒ Naturverständnis erhöhen

Einführung

👁️👁️ Große Klasse von Algorithmen

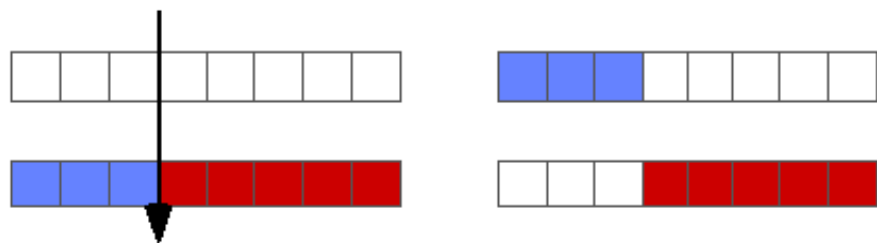
👁️👁️ Viele Anwendungsbereiche
⇒ Optimierung ist eine Hauptanwendung

👁️👁️ Suchheuristiken (Suchmethoden)

Das biologische Prinzip: Rekombination / Crossover

👁👁 Vorteil: Kurze Entwicklungszeit

👁👁 Kombinieren verschiedener Individuen



Quelle: [Hesser]

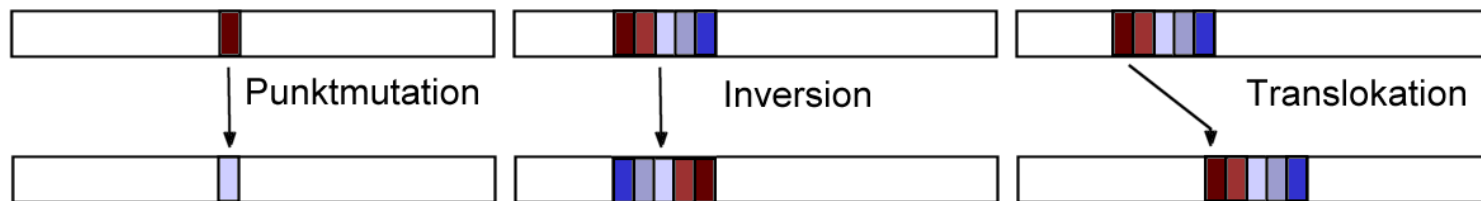
👁👁 n-Punkt-Crossover

👁👁 Hier: $n = 1$

Das biologische Prinzip: Mutation

👁👁 Vorteil: Völlig neue Varianten möglich

👁👁 Direkte Änderung eines Chromosoms



Schemazeichnung der Mutationsarten (nach Futuyama)

Quelle: [Hesser]

Das biologische Prinzip: Selektion + Generationenwechsel

👁️👁️ Fitness des Individuums

👁️👁️ Überleben der Starken

👁️👁️ Zeitfaktor

👁️👁️ Überleben

👁️👁️ Nur Nachkommen oder

👁️👁️ Eltern + Nachkommen kombiniert

Grundlagen

👁️👁️ Optimierung am Beispiel TSP

👁️👁️ Verschiedene Algorithmen

Travelling Salesman Problem



n Ziele



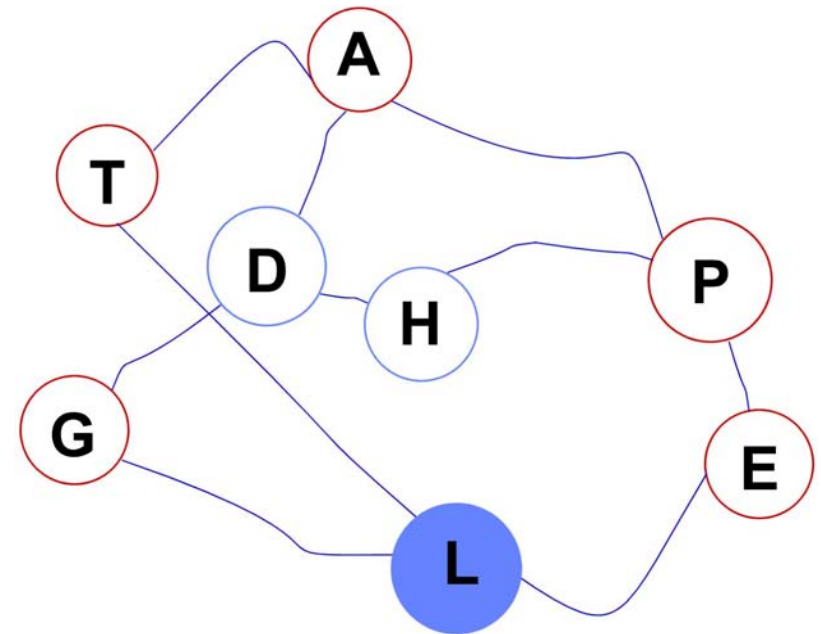
Jedes Ziel genau 1-mal besuchen



Ende = Ausgangsort



? Kürzester Rundweg ?



Anwendungen TSP

👁👁 Leiterplattenlayout

👁👁 Tourenplanung

👁👁 Alle Probleme...

👁👁 ...mit Reihenfolge...

👁👁 ...in diskreten Schritten

Verschiedene Algorithmen

 Randomisierte Suchverfahren

 Genetische Algorithmen

 Evolutionsstrategien

 Evolutionäre Programmierung

 Genetische Programmierung

Randomisierte Suchverfahren

Die Basis

Was brauchen wir?

👁👁 Suchraum S mit Nachbarschaft $N(x)$ für jeden Punkt $x \in S$

👁👁 Größe der Nachbarschaft z. B. beschränkt durch
Hamming-Abstand

$$H(x,y) = | \{ i \mid x_i \neq y_i \} |$$

👁👁 Fitnessfunktion $f: S \rightarrow M$ M geordnete Menge

Randomisierte Suchverfahren

Der Algorithmus

Wie geht's?

👁️👁️ Wähle Ausgangspunkt $x \in S$ zufällig

👁️👁️ Wähle $x' \in N(x)$ zufällig und berechne f

👁️👁️ Falls $f(x') > f(x)$, wähle x' als nächsten Ausgangspunkt
sonst fertig (Optimum in x gefunden)

Genetische Algorithmen

Worum geht's?

- 👁️👁️ Selektionsdruck durch probabilistische Selektion der Eltern
- 👁️👁️ Kindgeneration ersetzt Elterngeneration komplett
- 👁️👁️ Immer exakt so viele Kinder erzeugen, wie es Eltern gab
- 👁️👁️ Primärer Operator ist die Rekombination

Genetische Algorithmen

Wie geht's?

👁👁 Suchraum $\{0,1\}^n$

👁👁 Fitnessfunktion $f: S \rightarrow \mathbb{R}^+$ mit $F = f(x_1) + \dots + f(x_m)$
für m Individuen $x_i \in S$

👁👁 Wahrscheinlichkeit x_i zu wählen ist genau $f(x_i) / F$

👁👁 Crossover und Selektion (kaum oder keine Mutation)

Evolutionstrategien

Worum geht's?

👁️👁️ Genotyp ist reellwertig

👁️👁️ Eltern werden zufällig ausgewählt

👁️👁️ Deterministische Umweltselektion:
Die besten Individuen duplikatfrei auswählen
Bsp: n Eltern erzeugen $m > n$ Kinder,
daraus werden die n fittesten ausgewählt

👁️👁️ Primärer Operator ist die Mutation

Evolutionstrategien

Wie geht's?

👁️👁️ Anfangswerte festlegen \Rightarrow Elternteil

👁️👁️ Fitness messen

👁️👁️ Zufällige Mutation (d. h. Anfangswerte ändern)

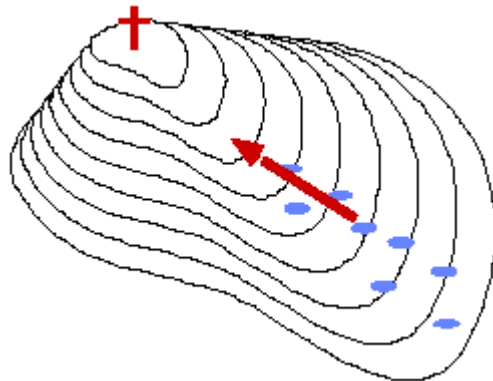
👁️👁️ Fitness messen: Nachkomme + Elternteil
Der bessere überlebt \Rightarrow Neues Elternteil

Funktionsprinzip ES

Wie funktioniert's?

👁️👁️ Gipfel ersteigen

👁️👁️ Starke Kausalität der monomodalen Zielfunktion



Quelle: [Hesser]

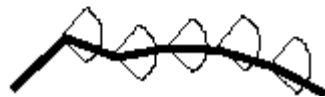
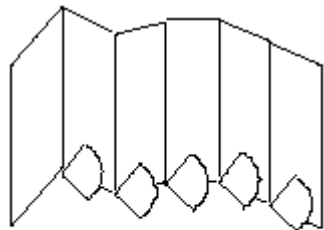
Starke Kausalität:

Kleine Änderungen der Systemparameter
⇒ Kleine Änderungen der Funktionswerte

Angewandte ES

👁👁 Anwendungsbeispiel Gelenkplatte

👁👁 Form optimieren bzgl. Strömungswiderstand



Quelle: [Hesser]

👁👁 5 Segmente

👁👁 Scharnierverbindung

👁👁 Messtafeln: Je 51 Winkelwerte

Evolutionäre Programmierung

- 👁👁 Verhalten, nicht Genetik
- 👁👁 Kein Rekombinationsoperator
- 👁👁 Keine Unterscheidung zwischen Genotyp und Phänotyp
- 👁👁 Auswahl der Besten aus Gesamtpopulation

Beispiel EP

👁👁 Endliche Automaten als Individuen

👁👁 Auswahlkriterium:
Welche Güte hat die Lösung der Aufgabe?

👁👁 Parameter werden in der nächsten
Generation zufällig geändert

Genetische Programmierung

- 👁👁 Ähnlich genetischen Algorithmen
- 👁👁 Hauptoperator ist Rekombination
- 👁👁 Mutation spielt untergeordnete Rolle

Evolutionäre Algorithmen und Neuronale Netze

- 👁👁 Lernen in NN bedeutet Optimierung:
Fehlerfunktion des Netzes wird minimiert
- 👁👁 Minimierung z. B. mit Evolutionären Algorithmen
- 👁👁 Netzparameter der Kinder unterscheiden sich
'zufällig' von denen der Eltern:
Wahrscheinlichkeit plus Zufallszahl

Weiterlesen im Web



[Hesser] Dr. Jürgen Hesser: "Evolutionäre Algorithmen".

Vorlesung Universität Mannheim 1996 / 2002

www-li5.ti.uni-mannheim.de/~hesser/EA/Ws0203/EAScript1.pdf

www-li5.ti.uni-mannheim.de/~hesser/EA/Ws0203/EAScript2.pdf



Prof. Dr. Ingo Wegener: "Evolutionäre Algorithmen"

Spezialvorlesung an der Universität Dortmund 2002

<http://ls2-www.cs.uni-dortmund.de/lehre/sommer2002/evol-alg/>

Weiterlesen auf Papier

👁👁 Marina Gorges-Schleuter: "Evolutionäre Algorithmen – Vorbild Natur". In: *Hubert B. Keller: "Maschinelle Intelligenz"*. vieweg 2000, ISBN 3-528-05489-1, € 34,-

👁👁 Wolfgang Banzhaf et. al.: "Genetic Programming – An Introduction". ca. 68,11 EUR.
Morgan Kaufmann Publishers 1998, ISBN 1-55860-510-X
dpunkt.verlag 1998, ISBN 3-920993-58-6