

Evolutionäre Algorithmen

Prof. Dr. habil. Wolfgang Kliesch

Minimierungsproblem

Ω ... Suchraum

$f : \Omega \mapsto \mathbb{R}$ Bewertungsfunktion

Gesucht: Menge der globalen Minimalpunkte

$$\mathcal{M} = \{x \in \Omega \mid f(x) \leq f(x') \quad \forall x' \in \Omega\}$$

Beispiel 1: Minimierung einer stetigen Funktion

$$\Omega = \mathbb{R}^n, \quad \mathbf{f} : \mathbb{R}^n \mapsto \mathbb{R}$$

Beispiel 2: Diskrete Approximation

Gegeben: Funktion $f : \mathbb{R} \mapsto \mathbb{R}$.

Gesucht Polynom

$$p_n(x) = \sum_{i=0}^n a_{n-i} x^{n-i},$$

so dass für vorgegebene Punkte x_1, \dots, x_m

$$\mathbf{f}(a_0, \dots, a_n) = \sqrt{\sum_{i=1}^m (f(x_i) - p_n(x_i))^2}$$

minimal.

Suchraum $\Omega = \mathbb{R}^{n+1}$,

Bewertungsfunktion: \mathbf{f} .

Beispiel 3: Problem des Handlungsreisenden (traveling salesman problem)

Gegeben sind n Städte, die paarweise durch Straßen verbunden sind.

$$G = (\mathcal{N}, \mathcal{E})$$

mit

$$\mathcal{N} = \{1, 2, \dots, n\}, \quad \mathcal{E} = \mathcal{N} \times \mathcal{N}.$$

\mathcal{P} ... Menge der Wege auf G .

Die Kostenfunktion

$$\gamma : \mathcal{E} \longmapsto \mathbb{R}$$

sei symmetrisch, d. h.

$$\gamma((s_i, s_j)) = \gamma((s_j, s_i)) \quad \forall (s_i, s_j) \in \mathcal{E}.$$

Suchraum

$$\Omega = \mathcal{S}_n,$$

\mathcal{S}_n ... Menge aller Permutationen von n Zahlen.

Bewertungsfunktion

$$\mathbf{f} : \mathcal{P} \longmapsto \mathbb{R}, \quad \mathbf{f}(p) := \sum_{(s_i, s_j) \in p} \gamma((s_i, s_j)),$$

$$p \in \mathcal{P}.$$

Grundbegriffe der Populationsgenetik

Eine Population ist die Gesamtheit aller Individuen einer Organismenart in einem begrenzten Gebiet.

- Ein *Gen* ist ein Abschnitt auf einem Chromosom, der die kleinste Einheit der genetischen Information darstellt.
- Das *Genom* ist die Gesamtheit aller Gene eines Organismus. Durch das Genom wird die *phänotypische Ausprägung* bestimmt.
- Ein Wert, den ein einzelnes Gen im Genom annehmen kann, wird als *Allel* bezeichnet.
- Die Gesamtheit des Genoms mit dem Phänotyp wird als *Individuum* bezeichnet.

- Die Gesamtheit aller Allele in einer Population wird *Genpool* genannt.
- Die Häufigkeit der Allele eines Gens in einer Population wird als *Genfrequenz* bezeichnet.

Natürliche Evolution

Evolution findet statt, wenn sich die Genfrequenz in einer Population ändert.

Evolutionsfaktoren

- *Rekombination*, findet bei der Paarung statt.
- *Mutation*, verursacht durch Fehler bei der Reproduktion der DNA.
- *Selektion*, bedingt durch Veränderung der Allelenhäufigkeit aufgrund unterschiedlich vieler Nachkommen der einzelnen Allele (Unterscheidung zwischen Eltern- und Umweltselektion).

Genfluss: Veränderung der Genfrequenz durch Zu-und Abwanderung von Individuen.

Gendrift: Veränderung der Genfrequenz durch Aussterben von Allelen einzelner Gene aufgrund von Zufallseffekten.

Evolution kann als ein Optimierungsverfahren aufgefasst werden, das durch Manipulation der Erbinformation von Organismen diese (optimal) an ihre Umwelt- und Lebensbedingungen anpasst. Damit kann die Evolution als Muster zur Konstruktion von Optimierungsverfahren dienen.

Simulierte Evolution

Eine Population von Lösungskandidaten, die als Individuen bezeichnet werden, werden einer simulierten Evolution unterworfen.

Grundschema der evolutionären Algorithmen

1. Bestimmung einer Ausgangspopulation
2. Bestimmung der Fitness der Individuen der Ausgangspopulation
3. Bestimmung der Individuen zur Erzeugung einer neuen Generation durch Selektion (Elternselektion)
4. Erzeugung einer neuen Generation von Individuen durch Rekombination
5. Erzeugung einer neuen Population aus der alten Population und der neuen Generation (Umweltselektion)

6. Veränderung einiger Individuen durch Mutation
7. Bestimmung der Fitness der Individuen der neuen Population
8. Beendigung falls Abbruchkriterium erfüllt, anderenfalls Fortsetzung mit Schritt 3

Beispiel 3 (traveling salesman problem)

$$\Omega = \mathcal{S}_6$$

Kostenfunktion

	1	2	3	4	5	6
1	0	5	8	11	4	7
2		0	10	4	9	12
3			0	6	17	8
4				0	6	5
5					0	11
6						0

Aufstellung der Operatoren

- Mutation
 - Vertauschung zweier Zahlen der Permutation
 - Inversion eines Teilstücks der Permutation
- Rekombination

Überlagere die beiden Elternwege zu einem orientierungslosen Graphen G und wähle als Nachfolgerknoten eines Knoten des Kindweges jeweils den Nachbarknoten aus dem Graphen G mit der kleinsten Anzahl von Nachfolgern.

Mutation (Inversion)

1. Eingabe: Individuum $A = (A_1, \dots, A_n)$
2. $B := A$
3. Bestimme Zufallszahlen $m, n \in \mathcal{N}$ so, dass $m < n$
4. $B_{m+j} = A_{n-j}, j = 0(1) \left(\left\lceil \frac{n-m+1}{2} \right\rceil - 1 \right),$
 $B_{n-j} = A_{m+j}, j = 0(1) \left(\left\lceil \frac{n-m+1}{2} \right\rceil - 1 \right)$
5. Ausgabe: B

Rekombination

1. Eingabe: Permutationen $A = (A_1, \dots, A_n)$
und $B = (B_1, \dots, B_n)$
2. $k := 1$
3. $i \leftarrow (k = A_i), j \leftarrow (k = B_j)$
 $\text{Adj}(k) = \{A_{(i+n-2) \bmod n+1}, A_{i \bmod n+1},$
 $B_{(j+n-2) \bmod n+1}, B_{j \bmod n+1}\}$
4. $k := k + 1$
5. $k \leq n \Rightarrow$ Schritt 4
6. Wähle $C_1 \in \{A_1, B_1\}$

7. $i := 1$

8. $\mathcal{C} := \{C_1, \dots, C_i\}$

9. $m := \min_{k \in \text{Adj}(C_i) \setminus \mathcal{C}} \{\#(\text{Adj}(k) \setminus \mathcal{C})\}$

10. Falls $m > 0$, setze

$\mathcal{K} := \{k \in \text{Adj}(C_i) \setminus \mathcal{C} \mid \#(\text{Adj}(k) \setminus \mathcal{C}) = m\},$
anderenfalls $\mathcal{K} = \emptyset$.

11. Wähle

$$C_{i+1} \in \begin{cases} \mathcal{K} & \text{falls } \mathcal{K} \neq \emptyset, \\ \mathcal{N} \setminus \mathcal{C} & \text{falls } \mathcal{K} = \emptyset \end{cases}$$

12. $i := i + 1$

13. $i \leq n \Rightarrow$ Schritt 8

14. Ausgabe: C

Algorithmus

1. Eingabe:
 F (Fitnessfunktion),
 n (Anzahl der Orte)
2. $t := 0$
3. Erzeuge Population $\mathcal{P}(t)$
 (mit 10 Individuen)
4. Bewerte die Individuen aus $\mathcal{P}(t)$
5. $\mathcal{P}' := \emptyset$
6. $i := 1$

7. Wähle Eltern $A, B \in \mathcal{P}(t)$
8. Falls Zufallszahl $u < 0.3$,
dann $C := \text{Recomb}(A, B)$
9. $C := \text{Mutation}(C)$
10. $\mathcal{P}' := \mathcal{P}' \cup \{C\}$
11. $i := i + 1$
12. $i \leq 40 \Rightarrow \text{Schritt 7}$
13. Bewerte alle $C \in \mathcal{P}'$
14. $t := t + 1$

15. Erzeuge $\mathcal{P}(t)$ aus den 10 besten Individuen aus $(\mathcal{P}' \cup \mathcal{P}(t-1))$

16. $t \leq 2000 \Rightarrow$ Schritt 5

17. Ausgabe: bestes Individuum aus $\mathcal{P}(t)$

Selektion

Die Selektion bewirkt im Wesentlichen eine Verschiebung der Häufigkeit, mit der die einzelnen Individuen in der Population auftreten.

Man unterscheidet

- *Elternselektion*
Sie bestimmt die Nachkommenpopulation.
- *Umweltselektion*
Sie bestimmt die nachfolgende Population und kann überlappend sein, d. h. nicht alle Individuen der Elternpopulation werden durch Individuen der Kinderpopulation ersetzt.

- *deterministische Selektion*

Individuen werden durch Fitnessfunktion bestimmt.

- *probalistische Selektion*

Individuen werden per Zufall ausgewählt.

- *duplikatfreie Selektion*

Jedes Individuum kann nur einmal ausgewählt werden.

Der Selektionsdruck wird mithilfe der
Selektionsintensität

$$I = \frac{\bar{F}_{sel} - \bar{F}}{\sigma}$$

gemessen, wobei

\bar{F} ... durchschnittliche Güte
vor der Selektion

\bar{F}_{sel} ... durchschnittliche Güte
nach der Selektion

σ ... Standardabweichung der Güte
vor der Selektion

deterministische Selektion

Sie erzielt den Selektionsdruck durch Verkleinerung der Population.

Es werden aus einer Population mit r Individuen die s , ($s < r$), besten Individuen für die nächste Generation ausgewählt.

a) (μ, λ) –Selektion (Komma–Selektion)

Es werden aus $r = \lambda$ Kindern $s = \mu$ Eltern für die nächste Generation ausgewählt. Je kleiner das Verhältnis λ/μ desto größer der Selektionsdruck (empfohlen: $\lambda/7 \leq \mu \leq \lambda/5$).

b) $(\mu + \lambda)$ –Selektion (Plus–Selektion)

Es werden aus $r = \lambda + \mu$ Eltern und Kindern $s = \mu$ Eltern für die nächste Generation ausgewählt.

probalistische Selektion

Bei der probalistischen Selektion kann für jedes Individuum ein Wert vorgegeben werden, der annähernd die Anzahl der Nachkommen bestimmt (*Fitnesswert*).

a) Roulette–Auswahlverfahren

- Bestimme Zufallszahl $z \in [0, 1]$
- Wähle $A_i \in \mathcal{P}(t)$ genau dann, wenn

$$\sum_{j=0}^{j < i} P(A_j) \leq z < \sum_{j=0}^{j \leq i} P(A_j)$$

i) *fitnessproportionale Selektion*
für Maximierungsprobleme

$$P(A_i) = \frac{F(A_i)}{\sum_{j=1}^r F(A_j)}$$

r = Anzahl der Individuen der Population

Selektionsintensität

$$I = \frac{\sigma}{\bar{F}(t)}.$$

Fitnesswerte = Gütewerte

Nachteil: Gibt es dominante Individuen in der Population, so wird sehr schnell die Diversität zerstört.

lineare Skalierung

$$\hat{F}(A_i) = \frac{F(A_i) - F_-^{(k)}}{F_+^{(k)} - F_-^{(k)}}$$

$F_-^{(k)}$... schlechtester Gütewert in den letzten k Populationen

$F_+^{(k)}$... bester Gütewert in den letzten k Populationen

ii) *rangbasierte Selektion*
für Minimierungsprobleme

$$P(A_i) = \frac{2}{r} \left(1 - \frac{i-1}{r-1} \right)$$

A_1 ... bestes Individuum der Population

A_r ... schlechtestes Individuum der Population

b) q -fache Turnierselektion

Es werden zufällig gleichverteilt q Individuen ausgewählt, die ein Turnier bestreiten. Das Individuum mit dem besten Gütewert gewinnt das Turnier und wird ausgewählt.

c) q -stufige zweifache Turnierselektion

Zunächst bestreitet jedes Individuum mit q zufällig gleichverteilt ausgewählten Gegnern jeweils ein Turnier. Im Fall des Sieges gibt es einen Punkt. Aus den Turnierergebnissen wird eine Rangliste aufgestellt. Die besten Individuen werden ausgewählt.

Genetischer Algorithmus

Binäre Kodierung

$$\mathbb{B}^k = \{0, 1\}^k$$

a) Kodierung

i) *standardbinäre Kodierung*

Eine binäre Zeichenkette $A = A_1 A_2 \dots A_k \in \mathbb{B}^k$ repräsentiert die reelle Zahl

$$\text{dec}(A) = g_u + \frac{g_o - g_u}{2^k - 1} \sum_{j=0}^{k-1} A_{k-j} 2^j$$

aus dem Intervall $[g_u, g_o]$.

Granularität: $\frac{g_o - g_u}{2^k - 1}$

dezimal	binär	Abstand
0	000	
1	001	1
2	010	2
3	011	1
4	100	3
5	101	1
6	110	2
7	111	1

ii) *Gray-Kodierung*

Aus der standardbinär kodierten Zeichenkette

$$A = A_1 A_2 \dots A_k$$

erhält man die Gray-kodierte Zeichenkette $B = B_1 B_2 \dots B_k$ mit

$$B_i = \begin{cases} A_i & \text{falls } i = 1 \\ A_{i-1} \oplus A_i & \text{sonst} \end{cases}$$

dezimal	Gray-kod.	Abstand
0	000	
1	001	1
2	011	1
3	010	1
4	110	1
5	111	1
6	101	1
7	100	1

Aus einer Gray-kodierten Zeichenkette

$$B = B_1 B_2 \dots B_k$$

erhält man die standardbinär kodierte Zeichenkette $A = A_1 A_2 \dots A_k$ mit

$$A_i = \bigoplus_{j=1}^i B_j.$$

b) Rekombination

- Ein–Punkt–Rekombination

- Wähle Zufallszahl z , $1 \leq z \leq n$.
- Tausche ab dem z -ten Gen der Eltern die Allele.

- Zwei–Punkt–Rekombination

- Wähle zwei Zufallszahl z_1, z_2 ,
 $1 \leq z_1, z_2 \leq n$.
- Tausche die Allele der Gene z_1 bis z_2 der Eltern.

- gleichmäßige Rekombination
 - Erzeuge ein Template (Muster)
 - Tausche die Allele eines Gens der Eltern genau dann, wenn das Template-Gen = 0.

c) Mutation

- Wähle zwei Zufallszahl z_1, z_2 mit $1 \leq z_1 \leq \mu$ und $1 \leq z_2 \leq n$.
- Invertiere das Allel des z_2 -ten Gens des z_1 -ten Individuums der Population.

Die Individuen der neuen Generation werden zu

- ca. 70% durch Crossover und
- ca. 30% durch Mutation von Individuen der Eltern–Generation

erzeugt.

Reelle Kodierung

a) Rekombination

- intermediäre Rekombination
Allel eines Kind-Gens = arithmetisches Mittel der Allele der entsprechenden Eltern-Gene.
- arithmetische Rekombination
Als Allel eines Kind-Gens wird ein Punkt auf der durch die Allele der entsprechenden Eltern-Gene bestimmten Gerade gewählt.

b) Mutation

Es wird mit einer gewissen Wahrscheinlichkeit auf jedes Allel eines Individuums ein Wert addiert.