

Evolutionäre Algorithmen

Kodierung, Fitness, Selektion

Prof. Dr. Rudolf Kruse **Pascal Held**

`{kruse,pheld}@iws.cs.uni-magdeburg.de`

Otto-von-Guericke-Universität Magdeburg

Fakultät für Informatik

Institut für Wissens- und Sprachverarbeitung

Übersicht

1. Kodierung

Hamming-Klippen

Problem der Epistasie

Verlassen des Suchraums

2. Fitness

3. Selektion

Wünschenswerte Eigenschaften einer Kodierung

nun: genauere Betrachtung der Elemente eines EAs

zunächst: **Kodierung der Lösungskandidaten**

- Kodierung ist problemspezifisch zu wählen
- kein allg. „Kochrezept“, um (gute) Kodierung zu finden
- aber: einige Prinzipien, die beachtet werden sollten

Wünschenswerte Eigenschaften einer Kodierung:

- Darstellung ähnliche Phänotypen durch ähnliche Genotypen
- ähnliche Fitness von ähnlich kodierten Lösungskandidaten
- Abgeschlossenheit von Ω unter verwendeten evolutionären Operatoren

Kodierung: erste wünschenswerte Eigenschaft

Darstellung ähnliche Phänotypen durch ähnliche Genotypen

- Mutationen einzelner Gene führen zu ähnlichen Genotypen (einzelne Alleländerungen → kleine Änderung des Chromosoms)
- wenn Eigenschaft nicht erfüllt ist, können naheliegende Verbesserungen u.U. nicht erzeugt werden
- Konsequenz: große Änderung des Genotyps, um zu ähnlichem (und u.U. besseren) Phänotyp zu kommen

Beispiel zur Verdeutlichung:

- Optimierung einer reellen Funktion $y = f(x_1, \dots, x_n)$
- Darstellung der (reellen) Argumente durch Binärcodes
- Problem: Kodierung als Binärzahl führt zu „Hamming-Klippen“

Binärkodierung reeller Zahlen

gegeben: reelles Intervall $[a, b]$ und Kodierungsgenauigkeit ε
gesucht: Kodierungsvorschrift für $x \in [a, b]$ als Binärzahl z ,
sodass z um weniger als ε von x abweicht

Idee: teile $[a, b]$ in gleich große Abschnitte der Länge $\leq \varepsilon$
 $\Rightarrow 2^k$ Abschnitte mit $k = \left\lceil \log_2 \frac{b-a}{\varepsilon} \right\rceil$
kodiert durch $0, \dots, 2^k - 1$

Binärkodierung reeller Zahlen

Abschnitte: $k = \left\lceil \log_2 \frac{b-a}{\varepsilon} \right\rceil$ oder $k = \left\lceil \log_2 \frac{b-a}{2\varepsilon} \right\rceil$

Kodierung: $z = \left\lfloor \frac{x-a}{b-a}(2^k - 1) \right\rfloor$ oder $z = \left\lfloor \frac{x-a}{b-a}(2^k - 1) + \frac{1}{2} \right\rfloor$

Dekodierung: $x = a + z \cdot \frac{b-a}{2^k - 1}$

Beispiel: Intervall $[-1, 2]$, Genauigkeit $\varepsilon = 10^{-6}$, $x = 0.637197$

$$k = \left\lceil \log_2 \frac{2 - (-1)}{10^{-6}} \right\rceil = \left\lceil \log_2 3 \cdot 10^6 \right\rceil = 22$$

$$z = \left\lfloor \frac{0.637197 - (-1)}{2 - (-1)}(2^{22} - 1) \right\rfloor = 2288966_{10}$$
$$= 1000101110110101000110_2$$

Hamming-Klippen

Problem:

- benachbarte Zahlen können sehr verschieden kodiert sein
- Kodierungen haben großen Hamming-Abstand (# verschied. Bits)
- Mutationen/Crossover überwinden „Hamming-Klippen“ schwer

Beispiel:

- Darstellung der Zahlen von 0 bis 1 durch 4-Bit-Zahlen
- also Abbildung $\frac{k}{15} \rightarrow k$
- $\frac{7}{15}$ (0111) und $\frac{8}{15}$ (1000) haben Hamming-Abstand 4

Gray-Kodes: Vermeidung von Hamming-Klippen

Lösung: Gray-Kodes

Hamming-Abstand benachbarter Zahlen = 1 Bit

binär	Gray
0000	0000
0001	0001
0010	0011
0011	0010

binär	Gray
0100	0110
0101	0111
0110	0101
0111	0100

binär	Gray
1000	1100
1001	1101
1010	1111
1011	1110

binär	Gray
1100	1010
1101	1011
1110	1001
1111	1000

Gray-Kodes: Berechnung

- **Gray-Kodes sind nicht eindeutig**
- jeder Kode, in dem sich Kodierungen benachbarter Zahlen nur in 1 Bit unterscheiden, heißt Gray-Kode
- Berechnung von Gray-Kodes meist aus Binärzahlkodierung

Häufigste Form:

$$\text{Kodierung: } g = z \oplus \left\lfloor \frac{z}{2} \right\rfloor$$

$$\text{Dekodierung: } z = \bigoplus_{i=0}^{k-1} \left\lfloor \frac{g}{2^i} \right\rfloor$$

\oplus : Exklusiv-Oder der Binärdarstellung

Gray-Kodes: Berechnung

Beispiel: Intervall $[-1, 2]$, Genauigkeit $\varepsilon = 10^{-6}$, $x = 0.637197$

$$z = \left\lfloor \frac{0.637197 - (-1)}{2 - (-1)} (2^{22} - 1) \right\rfloor = 2288966_{10}$$
$$= 1000101110110101000110_2$$

$$g = 1000101110110101000110_2$$
$$\oplus 100010111011010100011_2$$
$$= 1100111001101111100101_2$$

Gray-Kodes: Implementierung

```
unsigned int num2gray (unsigned int x)
{
    return x ^ (x >> 1);
} /* num2gray() */
```

```
unsigned int gray2num (unsigned int x)
{
    unsigned int y = x;
    while (x >>= 1) y ^= x;
    return y;
} /* gray2num() */
```

```
unsigned int gray2num (unsigned int x)
{
    x ^= x >> 16; x ^= x >> 8;
    x ^= x >> 4; x ^= x >> 2;
    return x ^ (x >> 1);
} /* gray2num() */
```

Kodierung: zweite wünschenswerte Eigenschaft

Ähnlich kodierte Lösungskandidaten sollten ähnliche Fitness haben.

Problem der Epistasie:

- *in der Biologie:* ein Allel eines (sog. epistatischen) Gens unterdrückt Wirkung aller möglichen Allele eines/mehrerer anderer Gene
- *in evolutionären Algorithmen:*
Wechselwirkung zwischen Genen eines Chromosoms, Änderung der Fitness durch Änderung eines Gens hängt stark von Ausprägungen der anderen Gene ab

Epistasie in der Biologie

Abweichungen von Mendel'schen Gesetzen beruhen oft auf Epistasie.

- Kreuzen von reinerbig schwarz- und weißsamigen Bohnen:
Anzahl schwarz- : weiß- : braunsamiger Bohnen = 12:1:3
(2. Kindgen.)
- d.h. Widerspruch zu Mendel'schen Gesetzen

Beispiel: Problem des Handlungsreisenden

Finde Rundreise mit minimalen Kosten durch n Städte.

zwei verschiedene Kodierungen der Rundreise:

1. Permutation der Städte

- besuche Stadt an k -ter Position im k -ten Schritt
- *geringe Epistasie*: z.B. Austausch zweier Städte ändert Fitness (Kosten) i.A. etwa gleich stark (lokale Tour-Änderung)

2. Angabe der Position der jeweils nächsten Stadt in Liste, aus der alle besuchten Städte gestrichen werden

- *hohe Epistasie*: Änderung eines Gens (speziell vorn liegendes) kann gesamte Rundreise ändern (globale Tour-Änderung)
- führt oft zu großen Änderungen der Fitness

Zweite Kodierung: Wirkung einer Mutation

Mutation	Chromosom	Liste noch zu besuchender Städte	Rundreise
davor	<div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 2px;">5</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 2px;">3</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 2px;">3</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 2px;">2</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 2px;">2</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 2px;">1</div>	1, 2, 3, 4, 5 , 6 1, 2, 3 , 4, 6 1, 2, 4 , 6 1, 2 , 6 1, 6 1	<div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 2px;">5</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 2px;">3</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 2px;">4</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 2px;">2</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 2px;">6</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 2px;">1</div>
danach	<div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 2px; background-color: #cccccc;">1</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 2px;">3</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 2px;">3</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 2px;">2</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 2px;">2</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 2px;">1</div>	1 , 2, 3, 4, 5, 6 2, 3, 4 , 5, 6 2, 3, 5 , 6 2, 3 , 6 2, 6 2	<div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 2px;">1</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 2px;">4</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 2px;">5</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 2px;">3</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 2px;">6</div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 2px;">2</div>

Epistasie: Zusammenfassung

- hoch epistatische Kodierung: keine Regelmäßigkeiten
- Mutation/Crossover führen zu fast zufälligen Fitnessänderungen
- Optimierungsproblem für EA schwer zu lösen
- sehr gering epistatische Kodierung: andere Verfahren oft besser

Kodierung: dritte wünschenswerte Eigenschaft

Der Suchraum Ω sollte (soweit möglich) unter den verwendeten evolutionären Operatoren abgeschlossen sein.

Verlassen des Suchraums ist u.U. Definitionsfrage
allgemein: **Suchraum wird verlassen**, wenn

- neues Chromosom nicht sinnvoll interpretiert/dekodiert wird
- Lösungskandidat bestimmte Anforderungen nicht erfüllt,
- Lösungskandidat durch Fitnessfunktion falsch bewertet wird

Problem der **Abstimmung** von Kodierung und EA-Operatoren:

- Verwenden kodierungsspezifischer evolutionärer Operatoren
- Einsatz von Mechanismen zur „Reparatur“ von Chromosomen
- Strafterms zur Verringerung der Fitness von Chromosomen $\notin \Omega$

Verlassen des Suchraums: Beispiel

n -Damen-Problem

zwei verschiedene Kodierungen: Chromosom der Länge n

1. Spaltenpositionen der Damen je Zeile (Allele $0, \dots, n - 1$)

Operatoren: Ein-Punkt-Crossover, Standardmutation

Entstehung stets gültiger Vektoren von Spaltenpositionen

⇒ Suchraum wird nicht verlassen

2. Nummern der Felder (Allele $0, \dots, n^2 - 1$) der Damen

Operatoren: Ein-Punkt-Crossover, Standardmutation

Entstehung von Chromosomen mit mehreren Damen auf 1 Feld

⇒ Suchraum wird verlassen

Verlassen des Suchraums: Lösungsansätze

n -Damen-Problem

- **Andere Kodierung verwenden:** Kodierung 1 vermeidet Problem und Ω deutlich kleiner (wenn durchführbar, beste Variante!)
- **Kodierungsspezifische evolutionäre Operatoren:**
 - *Mutation:* Ausschließen bereits vorhandener Allele beim Zufall
 - *Crossover:* suche zuerst Feldnummern je Chromosom, die im anderen Chromosom nicht vorkommen, und wende auf verkürzten Chromosomen Ein-Punkt-Crossover an
- **Reparaturmechanismus:** finde und ersetze mehrfach auftretende Feldnummern, bis alle Feldnummern verschieden
- **Strafterm:** verringere Fitness um Anzahl der Doppel-/Mehrfachbelegungen von Feldern, ggf. multipliziert mit Gewicht

Verlassen des Suchraums: am Beispiel des TSP

- Darstellung der Rundreise durch Permutation der Städte (Stadt an k -ter Position wird im k -ten Schritt besucht.)
- Ein-Punkt-Crossover kann Raum der Permutationen verlassen:

3	5	2	8	1	7	6	4
1	2	3	4	5	6	7	8

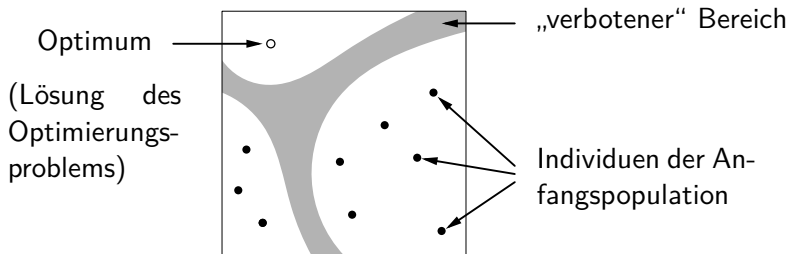
3	5	2	4	5	6	7	8
1	2	3	8	1	7	6	4

- **Kodierungsspezifische evolutionäre Operatoren:**
 - *Mutation*: z.B. Zweiertausch, Teilstück-Verschiebung, Inversion
 - *Crossover*: Kantenrekombination (wird später noch besprochen)
- **Reparaturmechanismus**: entferne doppelt auftretende Städte und hänge fehlende Städte ans Ende an:

3	5	2	4	5	6	7	8	1
---	---	---	---	--------------	---	---	---	---
- **Strafterm**: verringere Fitness um Wert c für jede fehlende Stadt

Verlassen des Suchraums

- falls Ω *nicht zusammenhängend*, erschwert *Reparaturmech.* Suche
- sofortiges Zurückführen „verbotener“ $x \in \Omega$ in erlaubte Bereiche



- in solchen Fällen: **Strafterm** einzuführen
- $x \in \Omega$ im „verbotenen“ Bereich wird bestraft aber nicht entfernt
- Strafterm sollte mit Zeit wachsen: unterdrückt in späteren Generationen $x \in \Omega$ in „verbotenen“ Bereichen

Übersicht

1. Kodierung

2. Fitness

Selektionsdruck

Selektionsintensität

Fitnessproportionale Selektion

Problem der vorzeitigen Konvergenz

Problem des verschwindenden Selektionsdrucks

Anpassung der Fitnessfunktion

3. Selektion

Prinzipien der Selektion

- bessere Individuen (bessere Fitness) sollen größere Chancen haben, Nachkommen zu zeugen (differentielle Reproduktion)
- **Selektionsdruck:** Stärke der Bevorzugung guter Individuen
- Wahl des Selektionsdrucks: Gegensatz von **Durchforstung des Suchraums (engl. exploration):**
 - möglichst breite Streuung der Individuen über Ω
 - möglichst große Chancen, globales Optimum zu finden⇒ geringer Selektionsdruck wünschenswert
- **Ausbeutung guter Individuen (engl. exploitation):**
 - Anstreben des (u.U. lokalen) Optimums in Nähe guter Individuen
 - Konvergenz zum Optimum⇒ hoher Selektionsdruck wünschenswert

Vergleich von Selektionsverfahren

Vergleichen von Verfahren für erzeugten Selektionsdruck durch Maße

- **Übernahmezeit:** ist Anzahl der Generationen bis Population konvergiert
(Population heißt *konvergiert*, wenn alle Individuen identisch)
- **Selektionsintensität:** bestimmt durch Selektionsdifferenzial zwischen durchschnittlicher Güte vor und nach Selektion

Selektionsintensität

nach [Weicker, 2007]

Definition (Selektionsintensität)

Sei (Ω, f, \succ) das betrachtete Optimierungsproblem und werde ein Selektionsoperator $\text{Sel}^\xi : (\mathcal{G} \times \mathcal{Z} \times \mathbb{R})^r \rightarrow (\mathcal{G} \times \mathcal{Z} \times \mathbb{R})^s$ auf eine Population P mit durchschnittlicher Güte μ_f und Standardabweichung σ_f der Gütwerte angewendet. Dann sei μ_f^{sel} die durchschnittliche Güte der Population P_{sel} und der Selektionsoperator besitzt die *Selektionsintensität*

$$I_{\text{sel}} = \begin{cases} \frac{\mu_f^{\text{sel}} - \mu_f}{\sigma_f} & \text{falls } \succ = >, \\ \frac{\mu_f - \mu_f^{\text{sel}}}{\sigma_f} & \text{sonst.} \end{cases}$$

Selektionsintensität

je größer I_{sel} , desto größer der Selektionsdruck

Beispiel:

- 10 Indiv. mit Fitness: 2.0, 2.1, 3.0, 4.0, 4.3, 4.4, 4.5, 4.9, 5.5, 6.0
- Selektion liefert Individuen mit Gütewerten: 2.0, 3.0, 4.0, 4.4, 5.5

$$\mu_f = \frac{1}{|P|} \sum_{i=1}^{|P|} A^{(i)} \cdot F \quad (\text{Mittelwert der Fitness})$$

$$\sigma_f = \sqrt{\frac{1}{|P| - 1} \sum_{i=1}^{|P|} (A^{(i)} \cdot F - \mu_f)^2} \quad (\text{Standardabweichung})$$

$$\Rightarrow \mu_f = 4.07, \quad \sigma_f = 1.27, \quad \mu_f^{sel} = 3.78, \quad I_{sel} = \frac{4.07 - 3.78}{1.27} = 0.228$$

Kritik an Selektionsintensität:

- dieses Maß setzt Standardnormalverteilung der Gütewerte voraus
- ist oft nicht der Fall bei allgemeinen Optimierungsproblemen

Wahl des Selektionsdrucks

- **beste Strategie:** zeitabhängiger Selektionsdruck
geringer Selektionsdruck in früheren Generationen
höherer Selektionsdruck in späteren Generationen
- d.h. zuerst gute Durchforstung des Suchraums,
dann Ausbeutung der erfolgversprechendsten Region
- Steuerung des Selektionsdrucks über Skalierung der
Fitnessfunktion oder über Parameter des Selektionsverfahrens
- wichtige **Selektionsverfahren:**
Glücksradauswahl, Rangauswahl, Turnierauswahl
- wichtige **Skalierungsmethoden:**
Anpassung der Fitnessvariation, linear dynamische Skalierung,
 σ -Skalierung

Glücksradauswahl (engl. roulette wheel selection)

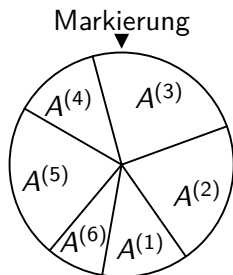
- das bekannteste Verfahren
- berechne relative Fitness der Individuen $A^{(i)}$, $1 \leq i \leq |P|$

$$f_{\text{rel}} \left(A^{(i)} \right) = \frac{A^{(i)}.F}{\sum_{j=1}^{|P|} A^{(j)}.F}$$

und interpretiere $f_{\text{rel}} \left(A^{(i)} \right)$ als Auswahlw'keit von $A^{(i)}$
(sog. **fitnessproportionale Selektion**)

- **Beachte:** absolute Fitness $A.F$ darf nicht negativ sein
(ggf. pos. Wert addieren oder neg. Werte zu Null setzen)
- **Beachte:** Fitness muss zu maximieren sein
(sonst: Selektion schlechter Individuen mit hoher W'keit)
- **Veranschaulichung:** Glücksrad mit 1 Sektor je Individuum $A^{(i)}$,
Sektorgrößen = relative Fitnesswerte $f_{\text{rel}} \left(A^{(i)} \right)$

Glücksradauswahl: Veranschaulichung



Auswahl eines Individuums:

1. drehe Glücksrad
2. wähle Indiv. des markierten Sektors

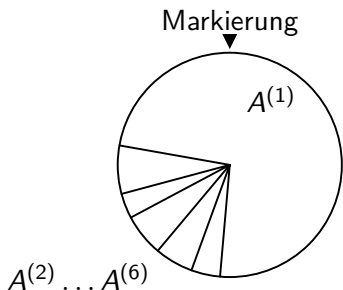
Auswahl der Zwischenpopulation:

- wiederhole Auswahl so oft, wie # Individuen in Population

technischer Nachteil: Berechnung der relativen Fitness durch Summierung aller Fitnesswerte (Normierungsfaktor)

- konstante Ausgangspopulation während Auswahl
- erschwerte Parallelisierung der Implementierung

Glücksradauswahl: Dominanzproblem



- Individuum mit sehr hoher Fitness kann Auswahl **dominieren**
- verstärkte Dominanz in Folgegenerationen aufgrund von Kopien/sehr ähnliche Individuen

⇒ **Crowding:** Population aus gleichen/sehr ähnlichen Individuen

- führt zum sehr schnellen Finden eines (lokalen) Optimums
- **Nachteil:** Diversität der Population geht verloren
 - Ausbeutung eines oder weniger guter Individuen
 - keine Durchforstung des Suchraums, sondern lokale Optimierung (in späten Generationen erwünscht, am Anfang unerwünscht)

Glücksradauswahl: Selektionsintensität

Satz

Bei reiner fitnessproportionaler Selektion in einer Population mit durchschnittlicher Güte μ_f und Gütevarianz σ_f^2 beträgt die Selektionsintensität

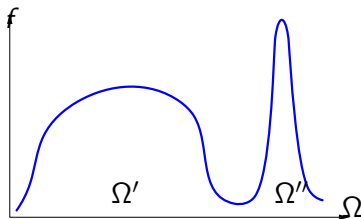
$$I_{sel} = \frac{\sigma_f}{\mu_f}.$$

- Beweis in Übungsaufgabe

Fitnessfkt: Problem der vorzeitigen Konvergenz

Dominanzproblem zeigt starken Einfluss der Fitnessfunktion auf Wirkung der fitnessproportionalen Selektion

- Problem der **vorzeitigen Konvergenz**:
falls Wertebereich der zu maximierenden Funktion sehr groß
- Beispiel: anfangs im Bereich Ω'' kein Chromosom \rightarrow Population bleibt durch Selektion nahe (lokales) Maximum im Bereich Ω'



Individuen, die sich dem Bereich zw. Ω' und Ω'' nähern, haben sehr schlechte Chancen auf Nachkommen

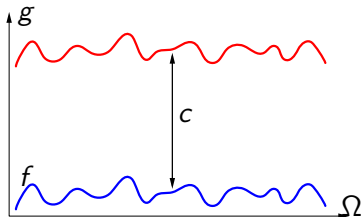
Problem des verschwindenden Selektionsdrucks

Problem der **absoluten Höhe** der Fitnesswerte i.V. zu ihrer **Variation**
umgekehrt: Problem des **verschwindenden Selektionsdrucks**:

- Maximierung von $f : \Omega \rightarrow \mathbb{R}$ ist äquivalent zur Maximierung von $g : \Omega \rightarrow \mathbb{R}$ mit $g(x) \equiv f(x) + c$, $c \in \mathbb{R}$

$$c \gg \sup_{x \in \Omega} f(x) \implies \forall x \in \Omega : g_{\text{rel}}(x) \approx \frac{1}{|P|} \quad (\text{Pop.-größe } |P|)$$

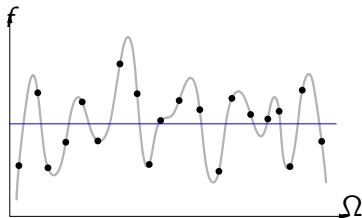
\Rightarrow (zu) geringer Selektionsdruck



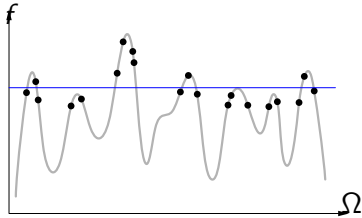
- obwohl Maxima an gleichen Stellen: mit EA unterschiedlich leicht zu finden
- mit g : nur (zu) geringe Unterschiede der f_{rel}

Problem des verschwindenden Selektionsdrucks

- EA erzeugt u.U. dieses Problem selbst
- er steigert tendenziell (durchschnittliche) Fitness der Individuen
- höherer Selektionsdruck am Anfang, da Fitnesswerte zufällig
- später: geringerer Selektionsdruck (umgekehrt wäre besser)
- Beispiel: Punkte zeigen Individuen der Generation



frühe Generation



späte Generation

Anpassung der Fitnessfunktion

Lösungsansatz: **Skalierung der Fitness** (Verschiebung der Fitnesswerte)

linear dynamische Skalierung:

$$f_{\text{lds}}(A) = \alpha \cdot A.F - \min \left\{ A^{(i)}.F \mid P(t) = \{A^{(1)}, \dots, A^{(r)}\} \right\}, \quad \alpha > 0$$

- Idee: Minimum der Fitness der Population bei allen Individuen abziehen
- anstatt Minimum von $P(t)$ auch Minimum der letzten k Generationen
- gewöhnlich $\alpha > 1$

σ -Skalierung:

$$f_{\sigma}(A) = A.F - (\mu_f(t) - \beta \cdot \sigma_f(t)), \quad \beta > 0$$

Problem: Wahl der Parameter

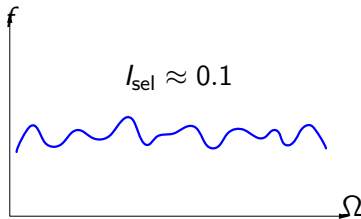
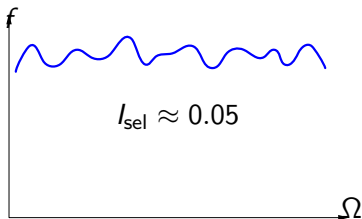
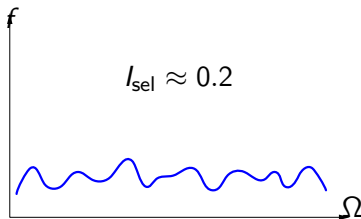
Anpassung der Fitnessfunktion

- betrachte **Variationskoeffizienten** der Fitnessfunktion

$$v = \frac{\sigma_f}{\mu_f} = \frac{\sqrt{\frac{1}{|\Omega|-1} \sum_{x' \in \Omega} \left(f(x') - \frac{1}{|\Omega|} \sum_{x \in \Omega} f(x) \right)^2}}{\frac{1}{|\Omega|} \sum_{x \in \Omega} f(x)}, \quad v(t) = \frac{\sigma_f(t)}{\mu_f(t)}$$

- empirische Feststellung: $v \approx 0.1$ liefert gutes Verhältnis von Durchforstung und Ausbeutung
 - wenn $v \neq 0.1$, dann Anpassung von f (z.B. durch Skalierung)
 - v ist nicht berechen-, sondern nur schätzbar
 - praktische Berechnungen von v : Ersetzen von Ω durch $P(t)$
 - somit: Annäherung von v durch **Selektionsintensität** $I_{sel}(t)$
- ⇒ in jeder Generation: berechne $I_{sel}(t)$ und passe f entsprechend an (σ -Skalierung mit $\beta = \frac{1}{I_{sel}^*}$, $I_{sel}^* = 0.1$)

Illustration der Selektionsintensität



- zu hohe I_{sel} : vorzeitige Konvergenz
- zu niedrige I_{sel} : verschwindender Selektionsdruck
- gut: $I_{\text{sel}} \approx 0.1$

Anpassung der Fitnessfunktion: Zeitabhängigkeit

- bestimme f_{rel} nicht direkt aus $f(x)$, sondern aus $g(x) \equiv (f(x))^{k(t)}$
- zeitabhängige Exponent $k(t)$ steuert Selektionsdruck
- **Verfahren zur Bestimmung von $k(t)$** [Michalewicz, 1996]
(soll Selektionsintensität I_{sel} in Nähe von $I_{\text{sel}}^* \approx 0.1$ halten)

$$k(t) = \left(\frac{I_{\text{sel}}^*}{I_{\text{sel}}} \right)^{\beta_1} \left(\tan \left(\frac{t}{T+1} \cdot \frac{\pi}{2} \right) \right)^{\beta_2} \left(\frac{I_{\text{sel}}}{I_{\text{sel}}^*} \right)^{\alpha}$$

I_{sel}^* , β_1 , β_2 , α : Parameter des Verfahrens

I_{sel} : Variationskoeffizient (z.B. aus $P(t=0)$ geschätzt)

T : maximale Anzahl zu berechnender Generationen

t : aktueller Zeitschritt (Nummer der Generation)

- Empfehlung: $I_{\text{sel}}^* = 0.1$, $\beta_1 = 0.05$, $\beta_2 = 0.1$, $\alpha = 0.1$

Anpassung der Fitnessfkt: Boltzmann-Selektion

- bestimme relative Fitness nicht direkt aus $f(x)$, sondern aus $g(x) \equiv \exp\left(\frac{f(x)}{kT}\right)$
- zeitabhängige **Temperatur** T steuert Selektionsdruck
- k ist Normierungskonstante
- Temperatur nimmt z.B. linear bis zu vorher festgelegten Maximalzahl an Generationen ab

Übersicht

1. Kodierung

2. Fitness

3. Selektion

- Glücksradauswahl

- Erwartungswertmodell

- Rangbasierte Selektion

- Turnierauswahl

- Elitismus

- Nischentechniken

- Charakterisierung

Glücksradauswahl: Varianzproblem

- Auswahl der Individuen ist zwar fitnessproportional, aber zufällig
- keine Garantie, dass gute Individuen in nächste Generation kommen, nicht einmal für bestes Individuum
- allg.: starke Abweichung (**hohe Varianz**) der Nachkommenzahl eines Individuums
- Berechnung des Erwartungswertes: siehe Übungsaufgabe
- sehr einfache, aber nicht unbedingt empfehlenswerte Lösung:

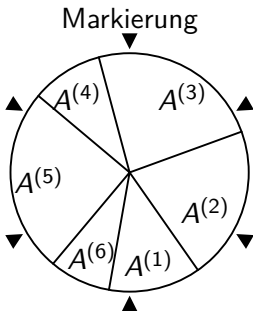
Diskretisierung des Fitnesswertebereichs

- berechne $\mu_f(t)$ und $\sigma_f(t)$ von P
- wenn $\mu_f(t) - \sigma_f(t) > f(x)$: 0 Nachkommen
- wenn $\mu_f(t) - \sigma_f(t) \leq f(x) \leq \mu_f(t) + \sigma_f(t)$: 1 Nachkomme
- wenn $f(x) > \mu_f(t) + \sigma_f(t)$: 2 Nachkommen

Erwartungswertmodell: Lsg. des Varianzproblems

- erzeuge für jeden Lösungskandidaten $\lfloor f_{\text{rel}}(s) \cdot |P| \rfloor$ Individuen
- fülle (Zwischen-)Population durch Glücksradauswahl auf

Alternative: **Stochastic Universal Sampling**



Auswahl der Zwischenpopulation:

- drehe Glücksrad einmal
- wähle ein Chromosom je Markierung
Hier:
 $1 \times A^{(1)}$, $1 \times A^{(2)}$, $2 \times A^{(3)}$, $2 \times A^{(5)}$.
- überdurchschnittlich gute Individuen kommen sicher in Zwischenpop.

Erwartungswertmodell: Varianten

Erzeugen der restlichen Individuen der (Zwischen-)Population durch

- Verfahren der **Wahlauswertung** (Mandats-/Sitzverteilung, z.B. größte Reste, Hare-Niemeyer, d'Hondt)
- **Glücksradauswahl** aber:
 - für jedes Individuum A mit 1 Nachkommen: $A.F' \leftarrow A.F - \Delta f$
 - falls $A.F' < 0$, dann keine weiteren Nachkommen von A
 - Prinzip für Wahl von Δf : bestes Individuum erhält höchstens festgelegte Zahl k von Nachkommen:

$$\Delta f = \frac{1}{k} \max\{A.F \mid A \in P(t)\}$$

Rangbasierte Selektion

1. sortiere Individuen nach ihrer Fitness absteigend:
jedes Individuum erhält **Rang** in Population
(aus Statistik: verteilungsfreie Verfahren, z.B. Rangkorrelation)
2. definiere W' keitsverteilung über Rangskala:
je kleiner Rangnummer, desto größer W' keit
3. Glücksradauswahl anhand der Verteilung

Vorteile:

- Vermeidung des Dominanzproblems: Fitnesswert beeinflusst Auswahlw'keit nicht direkt
- Steuerung des Selektionsdrucks durch W' keitsverteilung auf Rangskala

Nachteil: Sortieren der Individuen (Aufwand: $|P| \cdot \log |P|$)

Turnierauswahl

1. ziehe k Individuen ($2 \leq k < |P|$) zufällig aus $P(t)$
(mit oder ohne Zurücklegen, Auswahl *ohne* Berücksichtigung der Fitness, k sei **Turniergröße**).
2. Individuen tragen Turnier aus und bestes Individuum gewinnt:
Turniersieger erhält 1 Nachkommen in Zwischenpopulation
3. lege *alle* Individuen des Turniers (auch Sieger) in $P(t)$ zurück

Vorteile:

- Vermeidung des Dominanzproblems: Fitnesswert beeinflusst Auswahlw'keit nicht direkt
- Steuerung des Selektionsdrucks durch Turniergröße mit Einschränkungen

Modifikation: f_{rel} der Turnierteilnehmer bestimmt Gewinnw'keit
(Glücksradauswahl eines Individuums im Turnier)

Elitismus

- nur im Erwartungswertmodell (oder einer seiner Varianten) ist sichergestellt, dass bestes Individuum in Zwischenpopulation
 - wenn bestes Individuum in Zwischenpopulation: kein Schutz vor Veränderungen durch genetische Operatoren (auch beim Erwartungswertmodell)
- ⇒ Fitness des besten Individuums kann von einer Generation zur nächsten abnehmen (= unerwünscht)

Lösung: Elitismus

- *unveränderte* Übernahme des beste Individuum (oder der k , $1 \leq k < |P|$, besten Individuen) in nächste Generation
- *Elite* einer Population bleibt erhalten, daher *Elitismus*
- **beachte:** Elite ist *nicht* von normaler Auswahl ausgenommen: genetische Operatoren können sie verbessern

Elitismus

- oft: Kinder (Mutations-/Crossover-Produkte) ersetzen ihre Eltern
- „lokaler“ **Elitismus** (Elitismus zw. Eltern und Nachkommen)
 - *Mutation*: mutiertes Individuum ersetzt seinen Elter \leftrightarrow es hat mindestens so gute Fitness
 - *Crossover*: sortiere die vier beteiligten Individuen (2 Eltern, 2 Nachkommen) nach Fitness, beiden besten Individuen \rightarrow nächste Generation
- **Vorteil**: bessere Konvergenzeigenschaften, da lokales Optimum konsequenter angestrebt
- **Nachteil**: relativ große Gefahr des Hängenbleibens in lokalen Optima, da keine (lokalen) Verschlechterungen möglich

Nischentechniken: Vermeiden des Crowding

Deterministisches Crowding:

- *Idee*: erzeugte Nachkommen ersetzen stets ihnen ähnlichsten Individuen der Population

⇒ lokal weniger dicht besetztes Ω

- *benötigt*: Ähnlichkeits- oder Abstandsmaß für Individuen (bei binärkodierten Chromosomen z.B. Hamming-Abstand)

Variante des deterministischen Crowding:

- Crossover: bilde 2 Paare von Individuen (je 1 Elter, 1 Nachkommen), ordne 1 Nachkomme ähnlichsten Elter zu
- übernehme aus jedem Paar das bessere Individuum
- *Vorteil*: deutlich weniger Vergleiche zw. Individuen nötig (nur Teil der Pop. wird betrachtet)

Nischentechniken: Sharing

- *Idee*: reduziere Fitness eines Individuums, wenn in seiner Nachbarschaft noch weitere Individuen
anschaulich: **Individuen teilen sich Fitness einer Nische**
- *benötigt*: Ähnlichkeits- oder Abstandsmaß für Individuen
- *Beispiel*:

$$f_{\text{share}}(A) = \frac{A \cdot F}{\sum_{B \in P(t)} g(d(A, B))}$$

d : Abstandsmaß für Individuen

g : Wichtungsfunktion, die Form und Größe der Nische definiert,
z.B. sogenanntes **power law sharing**:

$$g(x) = \begin{cases} 1 - \left(\frac{x}{\varrho}\right)^\alpha & \text{falls } x < \varrho, \\ 0, & \text{sonst} \end{cases}$$

ϱ : Nischenradius, α : steuert Einflussstärke innerhalb Nische

Charakterisierung der Selektionsverfahren

statisch Auswahlw'keiten bleiben konstant

dynamisch Auswahlw'keiten ändern sich

extinctive (auslöschend) Auswahlw'keiten dürfen 0 sein

preservative (erhaltend) Alle Auswahlw'keiten müssen > 0

rein Individuen dürfen nur in 1 Generation Kinder haben

unrein Individuen dürfen in > 1 Generationen Kinder haben




rechts alle Individuen einer Pop. dürfen sich vermehren

links besten Individuen einer Population dürfen sich *nicht* vermehren (um vorzeitige Konvergenz zu vermeiden)

generational Eltermenge ist fest bis alle Nachkommen erzeugt

on the fly erzeugte Nachkommen ersetzen unmittelbar ihre Eltern

Literatur zur Lehrveranstaltung

-  Davidor, Y. (1990).
Lamarckian Sub-Goal reward in genetic algorithm.
In Proceedings of the European Conference on Artificial Intelligence (ECAI), page 189–194.
-  Michalewicz, Z. (1996).
Genetic Algorithms + Data Structures = Evolution Programs.
Springer-Verlag, New York, NY, USA, 3rd (extended) edition.
-  Weicker, K. (2007).
Evolutionäre Algorithmen.
Teubner Verlag, Stuttgart, Germany, 2nd edition.