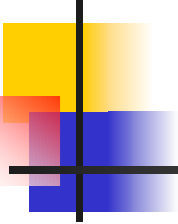


3. Prinzipien

Evolutionärer Algorithmen

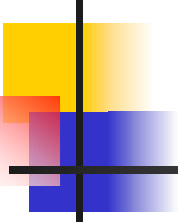
- ein klein wenig Theorie-

- 3.1 Selektionsmethoden
- 3.2 Evolutionäre Operatoren
- 3.3 Universelle Anwendbarkeit eines EAs?
- 3.4 Schema-Theorem und andere theoretische Resultate
- 3.5 Modelle für den Suchprozess



Ziel des Kapitels

- Bereitstellung allgemeingültiger Prinzipien der Arbeitsweise und des Entwurfs von EAs
- Schwierigkeiten der Parametereinstellung erläutern
- Grundsätzliche Selektionsmechanismen einführen
- Formale Beschreibung der Algorithmen von Operatoren
- Theoretische Überlegungen zur Leistungsfähigkeit von EAs
- Stochastische Suchprozesse vorstellen



Gliederung

3.1 Selektionsmethoden

3.2 Evolutionäre Operatoren

3.3 Universelle Anwendbarkeit eines EAs?

3.4 Schema-Theorem und andere
theoretische Resultate

3.5 Modelle für den Suchprozess



3.1 Selektionsmethoden

Selektion

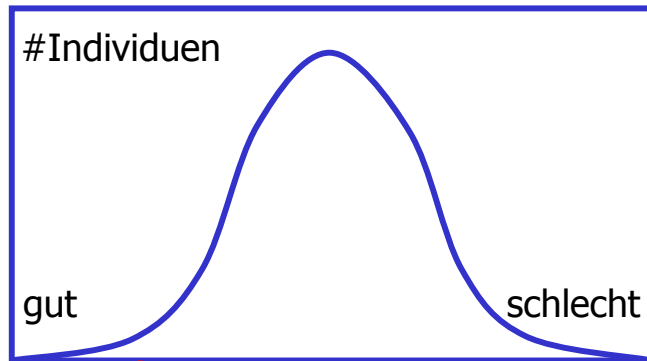
- gibt der Suche mit EAs eine gewisse Richtung vor
- Grundlage ist der Wert der Zielfunktion
- Vielfalt in einer Population wird durch Selektion reduziert
- Verbesserung der Güte einer Population durch Selektion der *besseren* Individuen

Zufallseinfluss bei der Selektion

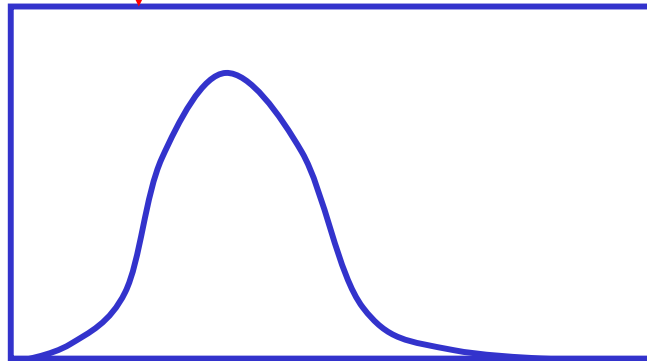
- Probabilistische Selektion, ohne Garantie der Übernahme der besten Individuen (duplikatfrei oder nicht duplikatfrei)
- Deterministische Selektion, mit garantierter Übernahme der besten Individuen (immer duplikatfrei)

Auswirkungen unterschiedlicher Selektionsansätze

Güteverteilung der Eltern

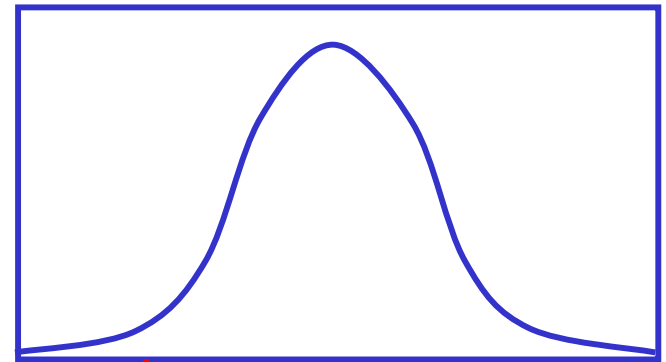


Selektionswahrscheinlichkeit
proportional zur Güte

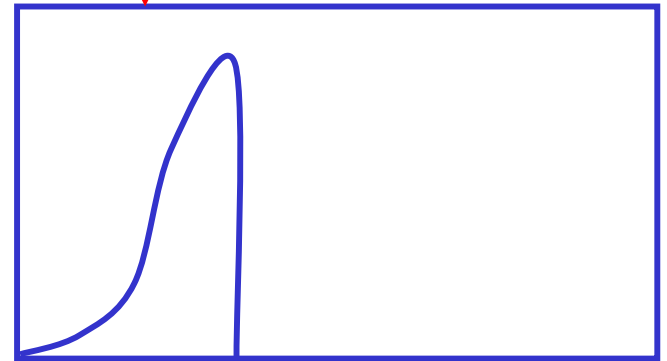


Güteverteilung der Kinder

Güteverteilung der Eltern



Deterministische Auswahl
der besten Individuen



Güteverteilung der Kinder



Eltern vs. Umweltselektion

- **Elternselektion:** Auswahl der Individuen, die sich reproduzieren sollen (*Heiratsschema*)
- **Umweltselektion:** Auswahl der Individuen, die in die nächste Generation übernommen werden (*Ersetzungsschema*)
- Selektionsmethode bestimmt den **Selektionsdruck**, dem Individuen ausgesetzt sind

Annahmen:

- Individuen seien nach Fitness geordnet, d.h. $P(t) = \{A^{(1)}, \dots, A^{(p)}\}$, dann gilt $A^{(i)}.F \geq A^{(j)}.F$ für $i < j$.
- Definiere $\text{Prob}(A^{(i)})$ als die Wahrscheinlichkeit das i -te Individuum auszuwählen, es gilt
$$\sum_{i=1}^p \text{Prob}(A^{(i)}) = 1$$



Bewertungsmaß für Selektionsmechanismen

Selektionsintensität in_S

- Durchschnittliche Güte der Population **vor** der Selektion: F
- Durchschnittliche Güte der Population **nach** der Selektion: F_{sel}
- σ ist Standardabweichung der Gütewerte vor der Selektion



$$in_S = \frac{(F_{sel} - F)}{\sigma}$$

Proportionale Selektion (nicht duplikatfrei)

$$\text{Prob}(A^{(i)}) = \frac{A^{(i)} \cdot F}{\sum_{j=1}^p A^{(j)} \cdot F}$$

	Population1		Population 2		Population 3	
i	A ⁽ⁱ⁾ .F	Prob(A ⁽ⁱ⁾)	A ⁽ⁱ⁾ .F	Prob(A ⁽ⁱ⁾)	A ⁽ⁱ⁾ .F	Prob(A ⁽ⁱ⁾)
1	1	1/15 ~ 0.067	1	~ 0.111	101	~ 0.196
2	2	2/15 ~ 0.133	1	~ 0.111	102	~ 0.198
3	3	3/15 ~ 0.2	1	~ 0.111	103	~ 0.2
4	4	4/15 ~ 0.267	1	~ 0.111	104	~ 0.202
5	5	5/15 ~ 0.333	5	~ 0.555	105	~ 0.204

1. Ausgewogene Verteilung
2. Individuum 5 dominiert
3. Sehr kleine Differenz zwischen bestem und schlechtestem Individuum

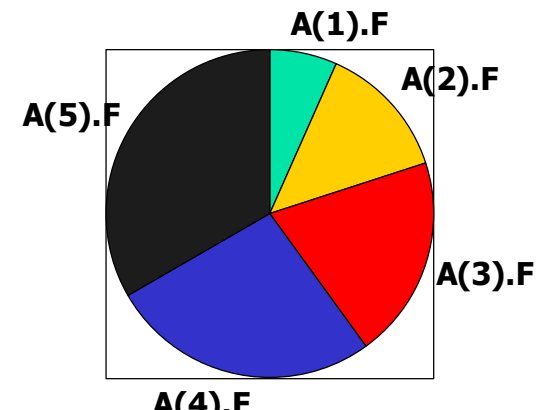
Roulette-Rad-Selektion

Proportionale Selektion (nicht duplikatfrei)

$$\text{Prob}(A^{(i)}) = \frac{A^{(i)} \cdot F}{\sum_{j=1}^p A^{(j)} \cdot F}$$

	Population1		Population 2		Population 3	
i	A ⁽ⁱ⁾ .F	Prob(A ⁽ⁱ⁾)	A ⁽ⁱ⁾ .F	Prob(A ⁽ⁱ⁾)	A ⁽ⁱ⁾ .F	Prob(A ⁽ⁱ⁾)
1	1	1/15 ~ 0.067	1	~ 0.111	101	~ 0.196
2	2	2/15 ~ 0.133	1	~ 0.111	102	~ 0.198
3	3	3/15 ~ 0.2	1	~ 0.111	103	~ 0.2
4	4	4/15 ~ 0.267	1	~ 0.111	104	~ 0.202
5	5	5/15 ~ 0.333	5	~ 0.555	105	~ 0.204

1. Ausgewogene Verteilung
2. Individuum 5 dominiert
3. Sehr kleine Differenz zwischen bestem und schlechtestem Individuum

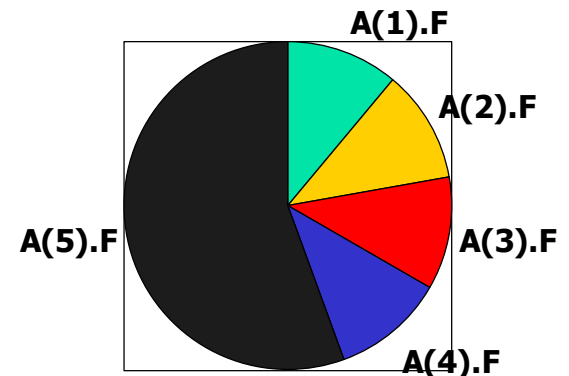


Proportionale Selektion (nicht duplikatfrei)

$$\text{Prob}(A^{(i)}) = \frac{A^{(i)} \cdot F}{\sum_{j=1}^p A^{(j)} \cdot F}$$

	Population1		Population 2		Population 3	
i	A ⁽ⁱ⁾ .F	Prob(A ⁽ⁱ⁾)	A ⁽ⁱ⁾ .F	Prob(A ⁽ⁱ⁾)	A ⁽ⁱ⁾ .F	Prob(A ⁽ⁱ⁾)
1	1	1/15 ~ 0.067	1	~ 0.111	101	~ 0.196
2	2	2/15 ~ 0.133	1	~ 0.111	102	~ 0.198
3	3	3/15 ~ 0.2	1	~ 0.111	103	~ 0.2
4	4	4/15 ~ 0.267	1	~ 0.111	104	~ 0.202
5	5	5/15 ~ 0.333	5	~ 0.555	105	~ 0.204

1. Ausgewogene Verteilung
2. Individuum 5 dominiert
3. Sehr kleine Differenz zwischen bestem und schlechtestem Individuum

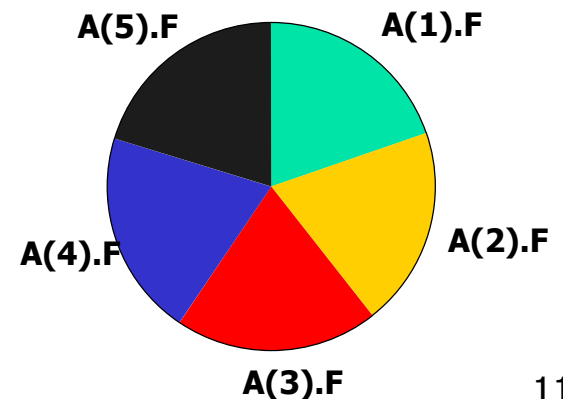


Proportionale Selektion (nicht duplikatfrei)

$$\text{Prob}(A^{(i)}) = \frac{A^{(i)} \cdot F}{\sum_{j=1}^p A^{(j)} \cdot F}$$

	Population1		Population 2		Population 3	
i	A ⁽ⁱ⁾ .F	Prob(A ⁽ⁱ⁾)	A ⁽ⁱ⁾ .F	Prob(A ⁽ⁱ⁾)	A ⁽ⁱ⁾ .F	Prob(A ⁽ⁱ⁾)
1	1	1/15 ~ 0.067	1	~ 0.111	101	~ 0.196
2	2	2/15 ~ 0.133	1	~ 0.111	102	~ 0.198
3	3	3/15 ~ 0.2	1	~ 0.111	103	~ 0.2
4	4	4/15 ~ 0.267	1	~ 0.111	104	~ 0.202
5	5	5/15 ~ 0.333	5	~ 0.555	105	~ 0.204

1. Ausgewogene Verteilung
2. Individuum 5 dominiert
3. Sehr kleine Differenz zwischen bestem und schlechtestem Individuum





Selektionsintensität:

Satz: Für die fitnessproportionale Selektion beträgt die Selektionsintensität in einer Population mit durchschnittlicher Güte $\bar{F}(t)$ zur Generation t und Gütevarianz σ^2 :

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

Beweis:

$$\begin{aligned} in_s &= \frac{1}{\sigma} \left(\left(\sum_{i=1}^p \text{Prob}(A^{(i)}) \cdot A^{(i)} \cdot F \right) - \bar{F}(t) \right) \\ &= \frac{1}{\sigma} \left(\left(\sum_{i=1}^p \frac{(A^{(i)} \cdot F)^2}{p \bar{F}(t)} \right) - \bar{F}(t) \right) = \frac{1}{\sigma} \left(\frac{1}{p \bar{F}(t)} \left(\sum_{i=1}^p (A^{(i)} \cdot F)^2 \right) - \bar{F}(t) \right) \\ &= \frac{1}{\sigma \bar{F}(t)} \left(\underbrace{\left(\frac{1}{p} \sum_{i=1}^p (A^{(i)} \cdot F)^2 \right) - \bar{F}(t)^2}_{\sigma^2} \right) \\ &= \frac{\sigma}{\bar{F}(t)} \end{aligned}$$

wg. $V[X] = E[X^2] - E[X]^2$

Selektionsintensitäten am Beispiel

	Population1		Population 2		Population 3	
i	A ⁽ⁱ⁾ .F	Prob(A ⁽ⁱ⁾)	A ⁽ⁱ⁾ .F	Prob(A ⁽ⁱ⁾)	A ⁽ⁱ⁾ .F	Prob(A ⁽ⁱ⁾)
1	1	1/15 ~ 0.067	1	~ 0.111	101	~ 0.196
2	2	2/15 ~ 0.133	1	~ 0.111	102	~ 0.198
3	3	3/15 ~ 0.2	1	~ 0.111	103	~ 0.2
4	4	4/15 ~ 0.267	1	~ 0.111	104	~ 0.202
5	5	5/15 ~ 0.333	5	~ 0.555	105	~ 0.204

- Population 1: $in_s \approx 0.471$
- Population 2: $in_s \approx 0.889$
- Population 3: $in_s \approx 0.014$

Varianten der proportionalen Selektion durch Skalierung

$$\text{Prob}(A^{(i)}) = \frac{A^{(i)} \cdot F - \text{worst}_j(A^{(j)} \cdot F)}{\sum_{j=1}^p (A^{(i)} \cdot F - \text{worst}_j(A^{(j)} \cdot F))}$$

Effekte:

- *schlechtestes* wird nie gewählt
- relative Unterschiede entscheiden

Alternativen: (Neuer Fitnesswert: $A^{(i)} \cdot F_{\text{neu}}$)

- Lineare Skalierung: $A^{(i)} \cdot F_{\text{neu}} = c_0 \cdot A^{(i)} \cdot F + c_1$, ($c_0, c_1 \in \mathbb{R}_+$)
- Logarithmische Skalierung: $A^{(i)} \cdot F_{\text{neu}} = c_0 - \log(A^{(i)} \cdot F)$
 - (c_0 ist so zu wählen, dass $A^{(i)} \cdot F_{\text{neu}}$ positiv ist)
- Exponentielle Skalierung:

$$A^{(i)} \cdot F_{\text{neu}} = (c_0 \cdot A^{(i)} \cdot F + c_1)^{c_2}, (c_0, c_1 \in \mathbb{R}_+, c_2 \in \mathbb{R})$$

■ ...

Skalierung hängt von den Daten ab!

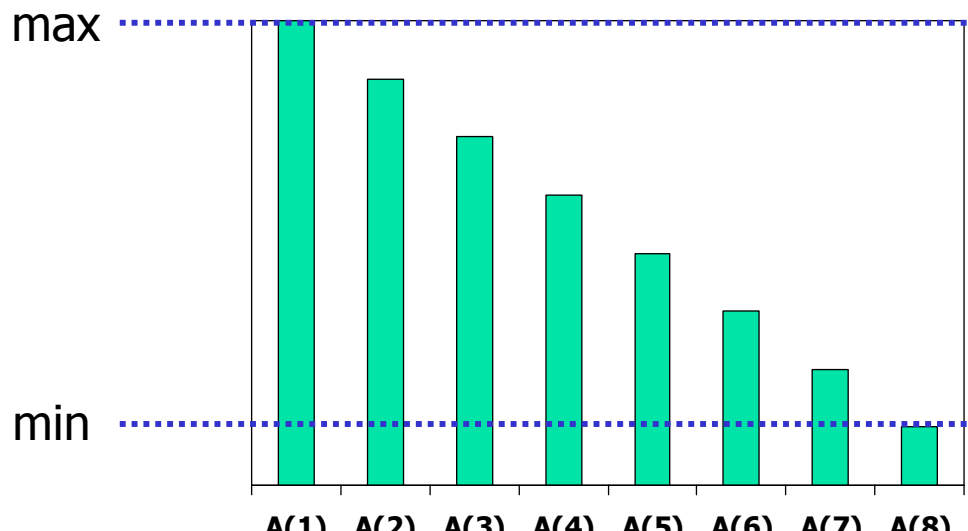
Rangbasierte Selektion (nicht duplikatfrei)

$$\text{Prob}(A^{(i)}) = \frac{1}{p} \left(\max - (\max - \min) \cdot \frac{i-1}{p-1} \right), 1 \leq i \leq p, \text{ mit } \max + \min = 2$$

Standardfall: $\max = 2$

Für den Standardfall gilt:

$$in_s \approx 0.564$$



- Rangbasierte Selektion ist unabhängig von der Verteilung der Fitnesswerte

Beispiel: max=2, min=0

$$\text{Prob}(A^{(i)}) = \frac{1}{p} \left(\max - (\max - \min) \cdot \frac{i-1}{p-1} \right), 1 \leq i \leq p$$

max = 2 $\Rightarrow$ min = 0:

$$\text{Prob}(A^{(i)}) = \frac{2}{p} \left(1 - \frac{i-1}{p-1} \right)$$

$$\text{Prob}(A^{(p)}) = \frac{2}{p} \left(1 - \frac{p-1}{p-1} \right) = 0 \quad \Rightarrow \text{schlechtestes Individuum}$$

$$\text{Prob}(A^{(1)}) = \frac{2}{p} \left(1 - \frac{1-1}{p-1} \right) = \frac{2}{p} \quad \Rightarrow \text{bestes Individuum}$$

für $p = 5$ erhält man dann:

$$\text{Prob}(A^{(1)}) = \frac{2}{5} = \frac{4}{10}$$

$$\text{Prob}(A^{(2)}) = \frac{3}{10}$$

$$\text{Prob}(A^{(3)}) = \frac{1}{5} = \frac{2}{10}$$

$$\text{Prob}(A^{(4)}) = \frac{1}{10}$$

$$= \frac{p-i}{\sum_{i=1}^p (i-1)} = \frac{p-i}{\frac{p(p-1)}{2}} = \frac{2}{p} \frac{(p-1) - (i-1)}{p-1} = \frac{2}{p} \left(1 - \frac{i-1}{p-1} \right)$$



Rang-proportionale Selektion

Weitere Beispiele

$$\text{Prob}(A^{(i)}) = \frac{1}{p} \left(\max - (\max - \min) \cdot \frac{i-1}{p-1} \right), 1 \leq i \leq p$$

$$\max = 1 \Rightarrow \min = 1:$$

$$\text{Prob}(A^{(i)}) = \frac{1}{p} \left(1 - 0 \cdot \frac{i-1}{p-1} \right) = \frac{1}{p}$$

$$p = 5 \Rightarrow \text{Prob}(A^{(i)}) = \frac{1}{5}$$

- Individuen werden unabhängig von Rang mit der gleichen Wahrscheinlichkeit ausgewählt

$$\max = \frac{3}{2} \Rightarrow \min = \frac{1}{2}:$$

$$\text{Prob}(A^{(i)}) = \frac{1}{p} \left(\frac{3}{2} - \frac{i-1}{p-1} \right)$$

$$p = 5 \Rightarrow$$

$$\text{Prob}(A^{(1)}) = \frac{1}{5} \left(\frac{3}{2} - \frac{0}{4} \right) = \frac{3}{10} = \frac{6}{20}$$

$$\text{Prob}(A^{(3)}) = \frac{1}{5} \left(\frac{3}{2} - \frac{2}{4} \right) = \frac{2}{10} = \frac{4}{20}$$

$$\text{Prob}(A^{(5)}) = \frac{1}{5} \left(\frac{3}{2} - \frac{4}{4} \right) = \frac{1}{10} = \frac{2}{20}$$



Die Parameter *min* und *max* bestimmen den **Verlauf** der Verteilungsfunktion

$$\text{Prob}(A^{(2)}) = \frac{1}{5} \left(\frac{3}{2} - \frac{1}{4} \right) = \frac{5}{20}$$

$$\text{Prob}(A^{(4)}) = \frac{1}{5} \left(\frac{3}{2} - \frac{3}{4} \right) = \frac{3}{20}$$



k -fach Turnierselektion (nicht duplikatfrei)

- Zufallsauswahl von $k < p$ Individuen aus der Population (oft $k=2$)
- Selektion des **besten** der k Individuen als Elternteil oder für die nächste Generation

➡ Turniersieger

Selektionswahrscheinlichkeit:

➡ Übung

Turnierselektion ist unabhängig von der Verteilung der Fitnesswerte.



Duplikatfreie Selektionen

- $(\mu + \lambda)$ – und (μ, λ) – Selektion
- Deterministische Methoden: Es werden die jeweils μ besten Individuen aus
 - einer Menge von λ Individuen $((\mu, \lambda)$ – Selektion), $\mu < \lambda$
 - einer Menge von $\mu + \lambda$ Individuen $((\mu + \lambda)$ – Selektion)ausgewählt
- μ Eltern erzeugen λ Nachkommen; die μ besten überleben
- Verhältnis $\frac{\mu}{\lambda}$ der Mengen (Population und Nachkommen) bestimmt den Selektionsdruck



Duplikatfreie probabilistische Selektionsverfahren

- K-stufige zweifache Turnierselektion
 - Durchführung von k-Turnieren zweier zufällig ausgewählter Individuen
 - Selektion der k Individuen mit den meisten Turniersiegen
- ...



Vergleich der Selektionsmechanismen

- Selektionsdruck hängt von den Parametern ab, ergibt sich ab er folgender Reihenfolge:
 1. Proportionale Selektion
 2. Rangbasierte Selektion
 3. Turnierselektion
 4. (μ, λ) – Selektion
 5. $(\mu + \lambda)$ – Selektion
- Heuristiken zur Parameterwahl existieren
- Hoher Selektionsdruck: Mutation ist notwendig um Diversität in der Population zu gewährleisten



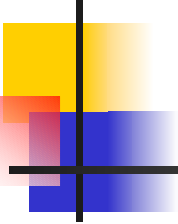
Einsatz der Selektionsmechanismen

- Selektionsmechanismen lassen sich zur **Eltern-** und **Umweltselektion** einsetzen
- Übliche Kombinationen
 - Genetische Algorithmen:
 - Elternselektion: probabilistisch/proportional
 - Umweltselektion: Identität, probabilistisch, deterministisch
 - Evolutionsstrategien:
 - Elternselektion: uniform
 - Umweltselektion: deterministische (μ, λ) – oder $(\mu + \lambda)$ – Selektion
 - Evolutionäres Programmieren:
 - Elternselektion: Identität
 - Umweltselektion: deterministisch, probabilistisch (duplikatfrei)



Effekte der Selektion

- Selektion verringert Diversität
 - Hoher Selektionsdruck fördert die Feinabstimmung (keine Erforschung neuer Gebiete)
 - Geringer Selektionsdruck fördert die Erforschung des Suchraums (minderwertige Individuen bleiben erhalten)
- ➡ Erhöhung des Selektionsdrucks zur Feinabstimmung nach dem in der Anfangsphase erforschten Suchraums



Gliederung

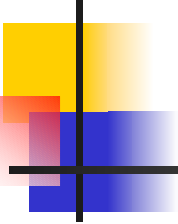
3.1 Selektionsmethoden

3.2 Evolutionäre Operatoren

3.3 Universelle Anwendbarkeit eines EAs?

3.4 Schema-Theorem und andere
theoretische Resultate

3.5 Modelle für den Suchprozess

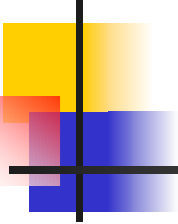


3.2 Evolutionäre Operatoren

- Mutation (unärer Operator) und
 - Rekombination (k-närer Operator)
- sind Gegenstücke zur Selektion

Ziel der Operatoren:

- Erzeugung neuer Individuen
- Erhöhung der Diversität in der Population
- Vorgabe der Suchrichtung



Rolle der Mutation

Effekt der Mutation hängt von der Struktur des Problems ab

- Bei *gutartigen* Problemen
(kleine Änderungen am Genotyp bewirken kleine Änderungen am Phänotyp bzw. Zielfunktion)
und kleiner Mutationswahrscheinlichkeit
→ hauptsächlich Feinabstimmung
- Bei *nicht gutartigen* Problemen gewährleistet Mutation Erreichbarkeit aller Punkte im Suchraum
- Bei großer Mutationswahrscheinlichkeit führt Mutation neue Diversität in Population ein (Makromutation)



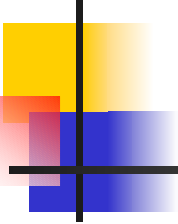
Nachbarschaften im Suchraum

- Mutationsoperator definiert eine Nachbarschaft
- Nachbarschaftsgraph definiert eine Erreichbarkeit in einem Schritt im Suchraum und auch lokale Optima

Definition: (Nachbarschaftsgraph)

Sei $M^\xi : G \rightarrow G$ ein Mutationsoperator, dann ist der Nachbarschaftsgraph $NG_M = (V, E)$ ein gerichteter Graph mit Knotenmenge $V = G$ und Kantenmenge

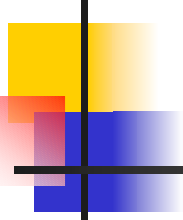
$$E = \left\{ (A_1, A_2) \mid A_1, A_2 \in G \wedge \exists_\xi : M^\xi(A_1) = A_2 \right\}$$



Lokales Optimum

Definition: (Lokales Optimum)

Sei $M^\xi : G \rightarrow G$ ein Mutationsoperator mit Nachbarschaftsgraph $NG_M = (V, E)$ und $F : G \rightarrow R$ eine induzierte Bewertungsfunktion. Dann induziert der Mutationsoperator ein lokales Optimum A , falls $F(A) \preceq F(A'), \forall A' \text{ mit } (A, A') \in E, \preceq \in \{<, >\}$.



Lokale Optima dieser Art sind zum Teil unabhängig von F und hängen ab vom

- Mutationsoperator und
- der gewählten Kodierung

Kodierung und Mutationsoperator führen u. U. neue lokale Optima ein

Ziel: Erzeuge möglichst wenige lokale Optima!

Probleme insbesondere bei stark strukturierten Genotypen, die durch Dekodierung auf Phänotypen abgebildet werden

➡ möglichst Einbettung der Nachbarschaft im phänotypischen Raum in die Nachbarschaft des genotypischen Raums



Beispiel

Dekodierung eines Funktionsoptimierungsproblems aus einem binären Suchraum

Standardbinärkodierung für das Intervall $[ug, og]$

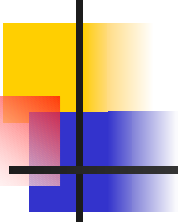
Binärstring $A_1, \dots, A_l$ kodiert

$$dec(A_1, \dots, A_l) = ug + \frac{og - ug}{2^l - 1} \sum_{i=0}^{l-1} A_{l-i} 2^i$$

$\frac{og - ug}{2^l - 1}$ bestimmt die Granularität der Darstellung

Standardbinärkodierung der Zahlen $0, \dots, 7$:

000, 001, 010, 011, 100, 101, 110, 111



Hamming-Distanz

- Übliches Abstandsmaß zweier binärer Werte

Definition: (Hamming-Distanz)

Zwei binäre Zeichenketten A und B der Länge l besitzen den Hamming-Abstand

$$d_H(A, B) = |\{i \mid A_i \neq B_i, 1 \leq i \leq l\}|$$

- gibt die Anzahl der Bits an, in denen zwei binäre Zeichenketten sich unterscheiden

Beispiel

- Standardbinärkodierung, $l=3$

- Maximaler Abstand z.B. zwischen 3_{10} und 4_{10}

$$d_H(011_2, 100_2) = 3 \quad \rightarrow \quad \text{Hamming-cliff}$$

Nachbarschaften werden missachtet!

- Alternative: Gray-Kodierung

Transformation eines (standardbinärkodierten) Binärstrings $A_1, \dots, A_l$ in *Gray*-kodierten Binärstring $B_1, \dots, B_l$:

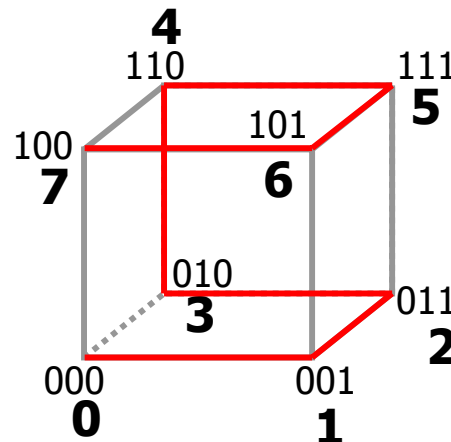
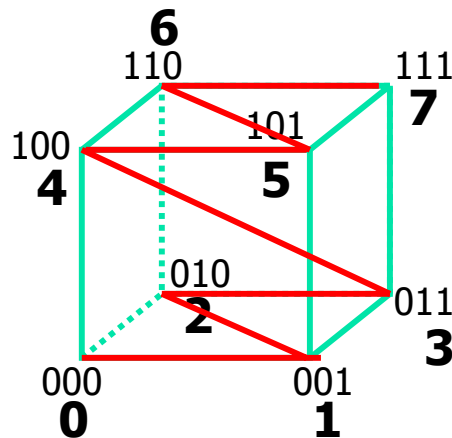
$$B_i = \begin{cases} A_i, i = 1 \\ A_{i-1} \oplus A_i, i > 1 \end{cases}$$

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

Gray-Kodierung der Zahlen 0,...,7:

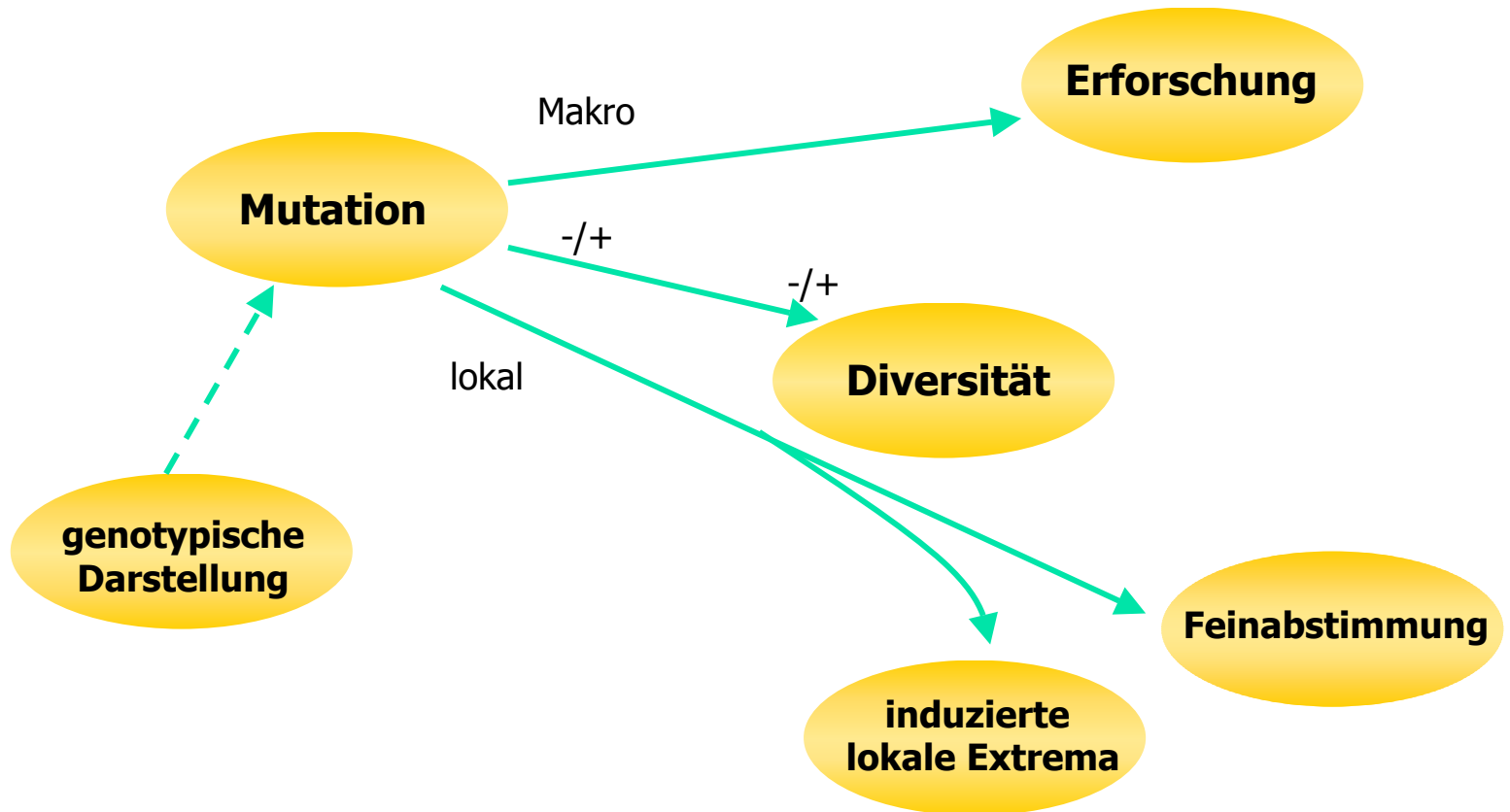
0	000	3	010	6	101
1	001	4	110	7	100
2	011	5	111		

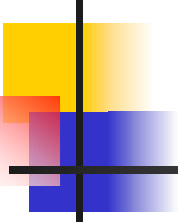
Nachbarschaftsgraphen der Kodierungen



Gray-Kodierung bettet Nachbarschaften im phänotypischen Raum in den genotypischen Raum ein. Allerdings können durch kleine Veränderungen am Genotypen großen Veränderungen des Phänotypen hervorgerufen werden!

Auswirkung der Mutation



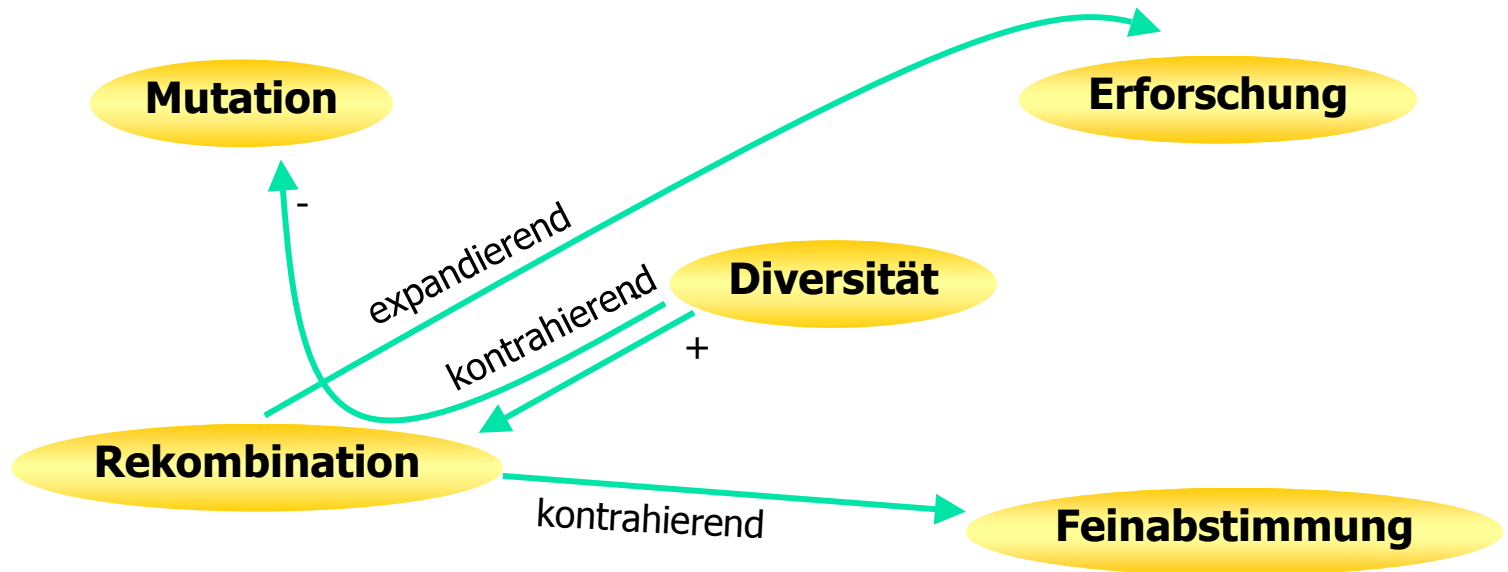


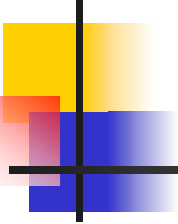
Rollen der Rekombination

- Nachkommen entstehen durch **Rekombinieren** (=Mischen) der Eigenschaften der Elternindividuen
 - ➡ es entstehen keine neue Informationen bzw. Allele
 - ➡ Vorteilhafte Eigenschaften sollen zusammengeführt werden
- Diversität in der Population ist erforderlich
 - ➡ Konvergenz
- Ansätze:
 - **Kombinierender Operator:** Eigenschaften mehrere Individuen werden kombiniert
 - **Expandierender Operator:** Kombination von Eigenschaften in Abhängigkeit von Gütewerten der Individuen (bzw. Gene der Individuen) ➡ problemspezifisch

Auswirkungen der Rekombination

- Integration der Rekombination in einen Algorithmus ist oft schwieriger als die Integration der Mutation
 - Ziel der Rekombination
 - Gültige Nachkommen





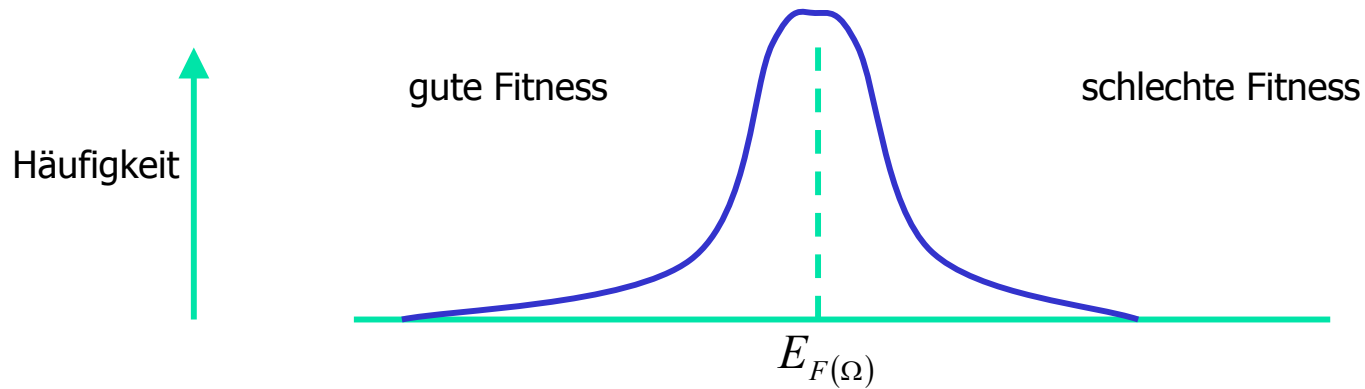
Bedeutung der Diversität

- Vielfalt in einer Population
- Bisher keine eindeutige Definition
 - Üblich ist Aufsummieren der paarweisen Entfernungen der Individuen einer Population
 - Oder die vorhandene Anzahl aller Allele pro Gen
- Diversität ist notwendig um den Suchraum zu erforschen
 - Vermeidung der Konvergenz zu einem lokalen Optimum
- Reduktion der Diversität durch Selektion
- Erhöhung der Diversität durch
 - (Makro-)Mutation
 - Rekombination
- Hohe Diversität ist notwendig für erforschende Eigenschaften der Rekombination

Zufallseinfluss und Verfahrenssteuerung

Überlegungen am einem stark idealisierten Beispiel:

Sei die Güteverteilung der Individuen normalverteilt über dem Suchraum - hier Darstellung bei linearisiertem Suchraum Ω



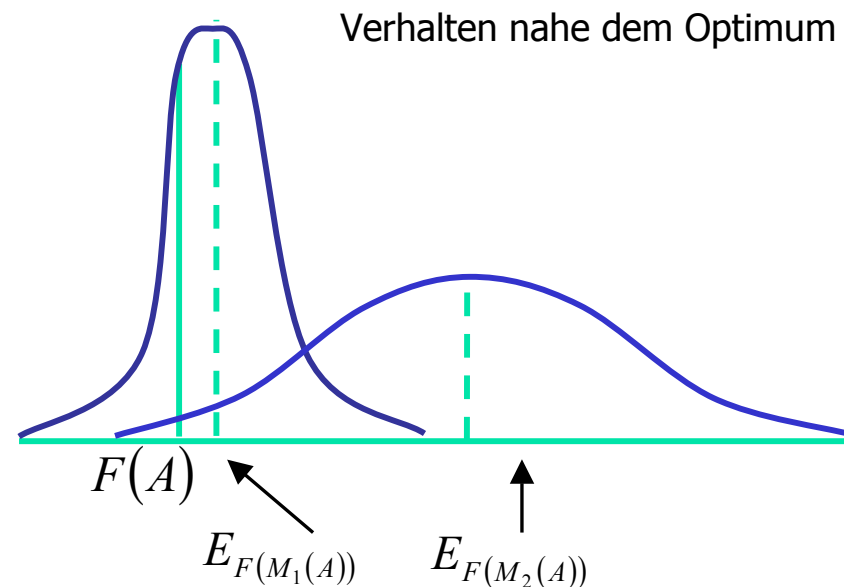
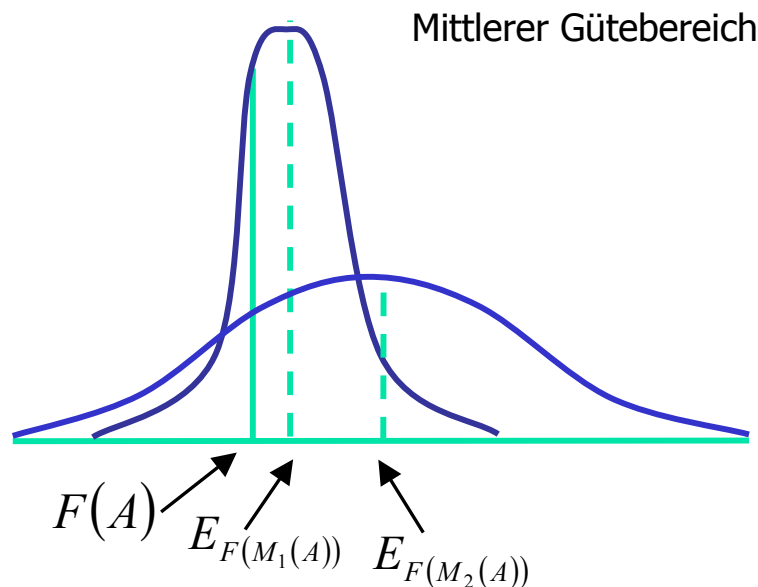
Untersuchung eines fiktiven Mutationsoperators, die Nachfolger bzgl. obiger Güteverteilung erzeugen

Freiheitsgrad bei der Auswahl des Mutationsoperators:

Varianz der Güte des Operators

- Kleine Gütevarianz: lokaler Operator (M_1)
- Große Gütevarianz: zufälliger Operator (M_2)

Erwartungswert der Fitness nach Anwendung des Operators hängt von der Fitness des Individuums ab



Bewertung der Mutationsoperatoren

- die Qualität von M ist abhängig von A
- ein lokaler Operator ist nahe am Optimum besser
- ein zufälliger Operator ist im Bereich mittlerer Güte besser

➡ Zufallseinfluss sollte im Laufe der Optimierung abnehmen

- **Statische Anpassung** durch Veränderung der Operatoren oder Parameter nach fester Anzahl von Experimenten (Auswertungen)
 - garantiert nicht, dass Veränderungen zum Suchergebnis passen
- **Adaptive Anpassung** durch Interpretation der Suchergebnisse
 - Qualität hängt ab von Formulierung der Anpassungsregeln
- **Selbstadaptive Anpassung** durch Speicherung der Verfahrensparameter in den Strategieparametern der Individuen und *evolutionäre Veränderung* der Strategieparameter
 - Aufwändige aber vielversprechende Methode



Gliederung

3.1 Selektionsmethoden

3.2 Evolutionäre Operatoren

3.3 Universelle Anwendbarkeit eines EAs?

3.4 Schema-Theorem und andere
theoretische Resultate

3.5 Modelle für den Suchprozess



3.3 Universelle Anwendbarkeit eines Optimierers

Im Gegensatz zu den meisten deterministischen Optimierungsmethoden stellen randomisierte und evolutionäre Algorithmen keine zusätzlichen Anforderungen an das zu lösende Problem

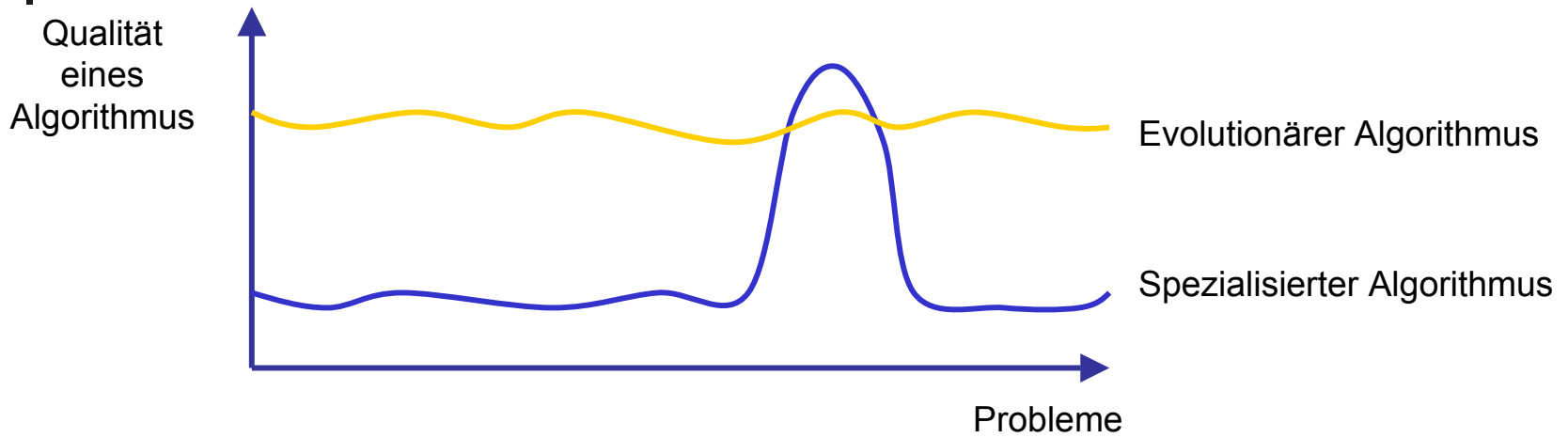
→ Universelle Optimierer (???)

Was kann man von einem Algorithmus erwarten, der keine weiteren Kenntnisse von dem zu lösenden Problem hat?

Um auf diese Frage eine Antwort zu finden, muss die Klasse der zu betrachtenden Probleme eingeschränkt werden!

Wir betrachten Probleme auf einem endlichen Suchraum Ω (nicht unrealistisch, da Zahl der Speicherzustände endlich)

Die frühen Hoffnungen und das No Free Lunch-Theorem



Die vermutete Überlegenheit von EAs

Sei

- 
- $\Omega = \{x_1, \dots, x_m\}$ der Suchraum
 - F die Menge der möglichen Optimierungsprobleme, so dass alle $f \in F$ von der Form $f : \Omega \rightarrow \mathbb{R}$ sind
 - $Alg : F \times N \rightarrow \Omega^n$ ein Algorithmus, der angewendet auf das Problem f , n Elemente aus Ω betrachtet

Also $Alg(f, n) = (y_1, \dots, y_n) \in \Omega^n$

(Annahme: alle y_i unterschiedlich)

- $Q(Alg(f, n))$ die Qualität der gefundenen Lösung
$$Q(Alg(f, n)) = H_n(f(y_1), \dots, f(y_n))$$

für eine Funktion H_n

- Der Erwartungswert der Qualität von Alg ist definiert als

$$E[Q(Alg(f, n))] = \frac{1}{|F|} \sum_{f \in F} Q(Alg(f, n))$$

- Alg_1 ist besser als Alg_2 , falls

$$E[Q(Alg_1(f, n))] > E[Q(Alg_2(f, n))]$$

No Free Lunch Theorem

Satz 3.1 (NFL)

Für zwei beliebige Algorithmen Alg_1 und Alg_2 , die jeweils n unterschiedliche Elemente aus dem Suchraum Ω betrachten, gilt:

$$E[Q(Alg_1(f, n))] = E[Q(Alg_2(f, n))]$$

Beweisskizze: (bei einigen vereinfachenden Annahmen)

Seien $r_1, \dots, r_m$ die Gütewerte der Elemente aus Ω . Jedes f lässt sich durch eine Permutation π aus S_m realisieren, da $F(x_i) = r_{\pi(i)}$

- ➡ es gibt $m!$ Funktionen in F
- ➡ es existieren $m * \dots * (m - n + 1)$ Argumentkombinationen für H_n (*Stichprobe in Reihenfolge ohne Zurücklegen*)
- ➡ jede Argumentkombination wird von $(m-n)!$ Funktionen aus F angenommen



Welche Gütewerte treten bei welchen Funktionen aus F für einen festen Algorithmus Alg auf?

Wir zeigen, dass für beliebige Gütewerte

$$r_{j_1}, \dots, r_{j_k} \quad (j_l \neq j_m \text{ falls } l \neq m)$$

genau für $(m-k)!$ Funktionen $f \in F$ gilt:

$$F(\text{Alg}(F, k)_1) = r_{j_1}$$

$$\vdots$$

$$F(\text{Alg}(F, k)_k) = r_{j_k}$$

(Induktionsbeweis)

Für $k=1$ wählt der Algorithmus einen Lösungskandidaten $y_1 \in \Omega$

Für eine beliebige Funktion $f \in F$ gilt mit Wahrscheinlichkeit $\frac{1}{m}$

$$f(y_1) = r_{j_1} \Rightarrow \text{es gibt } (m-1)! \text{ verschiedene } f \text{ dieser Art}$$

$\Rightarrow$ Aussage gilt für $k=1$



Für $k \rightarrow k + 1$:

- Algorithmus hat bereits die Werte $y_1, \dots, y_k$ mit Gütefunktionswerten $r_{j_1}, \dots, r_{j_k}$ gewählt
- Für eine zufällig aus F gewählte Funktion f wird mit Wahrscheinlichkeit $\frac{1}{m-k}$ Wert $r_{j_{k+1}}$ aus den verbleibenden Werten gewählt $\Rightarrow$ es gibt $(m-k-1)!$ Funktionen, die bei jeder beliebigen Wahl von $r_{j_{k+1}}$ den Wert annehmen

Die Definition von $Q(Alg)$ zeigt, dass genau $m * \dots * (m - n + 1)$ Argumentkombinationen für H_n existieren und jede dieser Argumentkombinationen von $(m-n)!$ Funktionen angenommen wird (d.h. alle sind gleich wahrscheinlich)

- ➡ die Werte der Summe zur Berechnung der Erwartungswerte sind gleich für alle Algorithmen
- ➡ damit sind auch die Erwartungswerte gleich

Bezüglich der Klasse aller Probleme verhält sich jeder Algorithmus gleich!



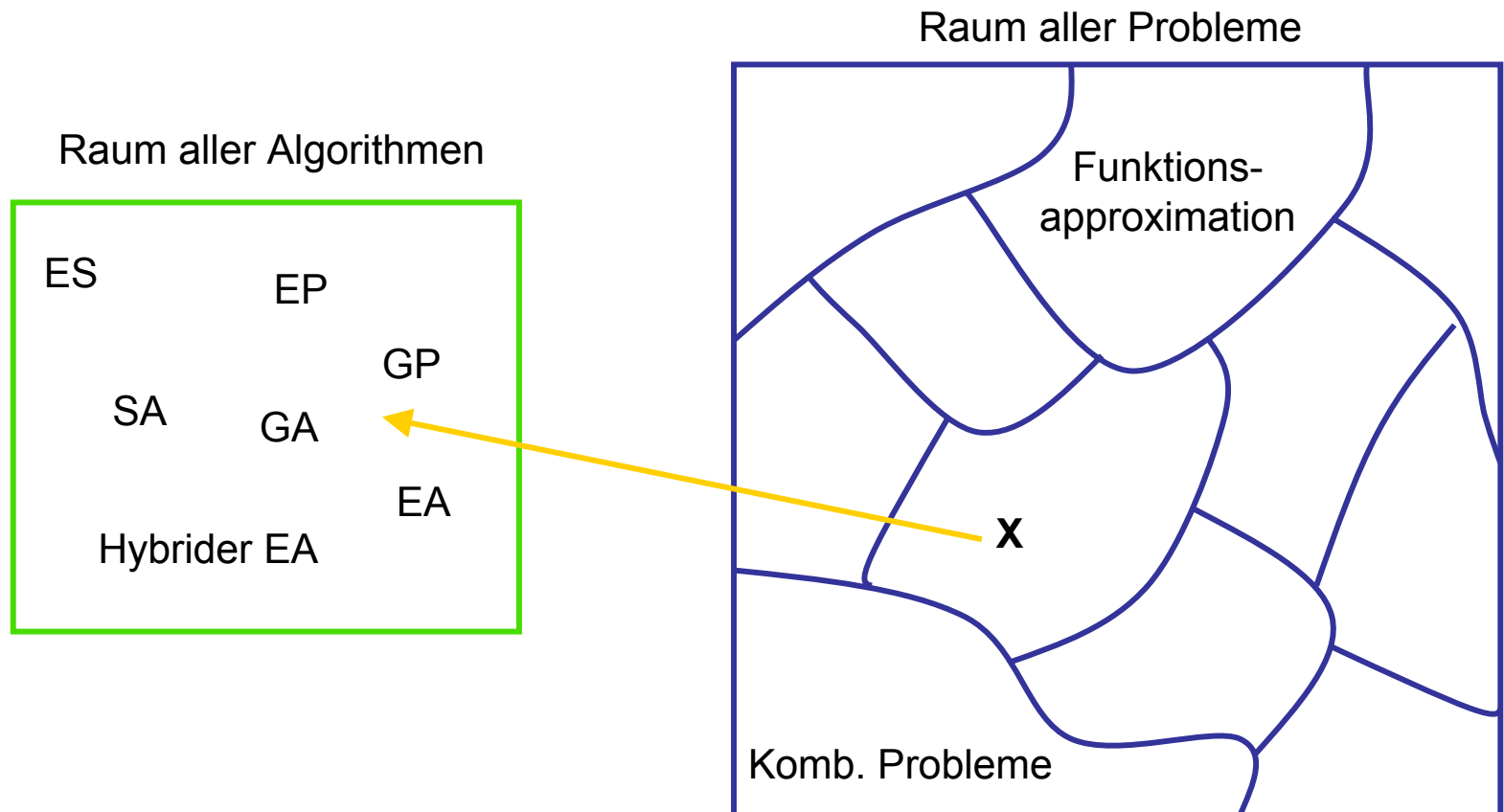
Verbesserungen sind nur möglich, wenn zusätzliche Kenntnisse über das Problem vorhanden sind

- Sei bekannt, dass ein Problem zur Klasse $F' \subset F$ gehört
- Gesucht wird Alg^* mit $E[Q(Alg^*(F', n))] > E[Q(Alg_i(F', n))]$ für alle (oder die meisten) anderen Algorithmen Alg_i . Dies bedeutet gleichzeitig (no free lunch):

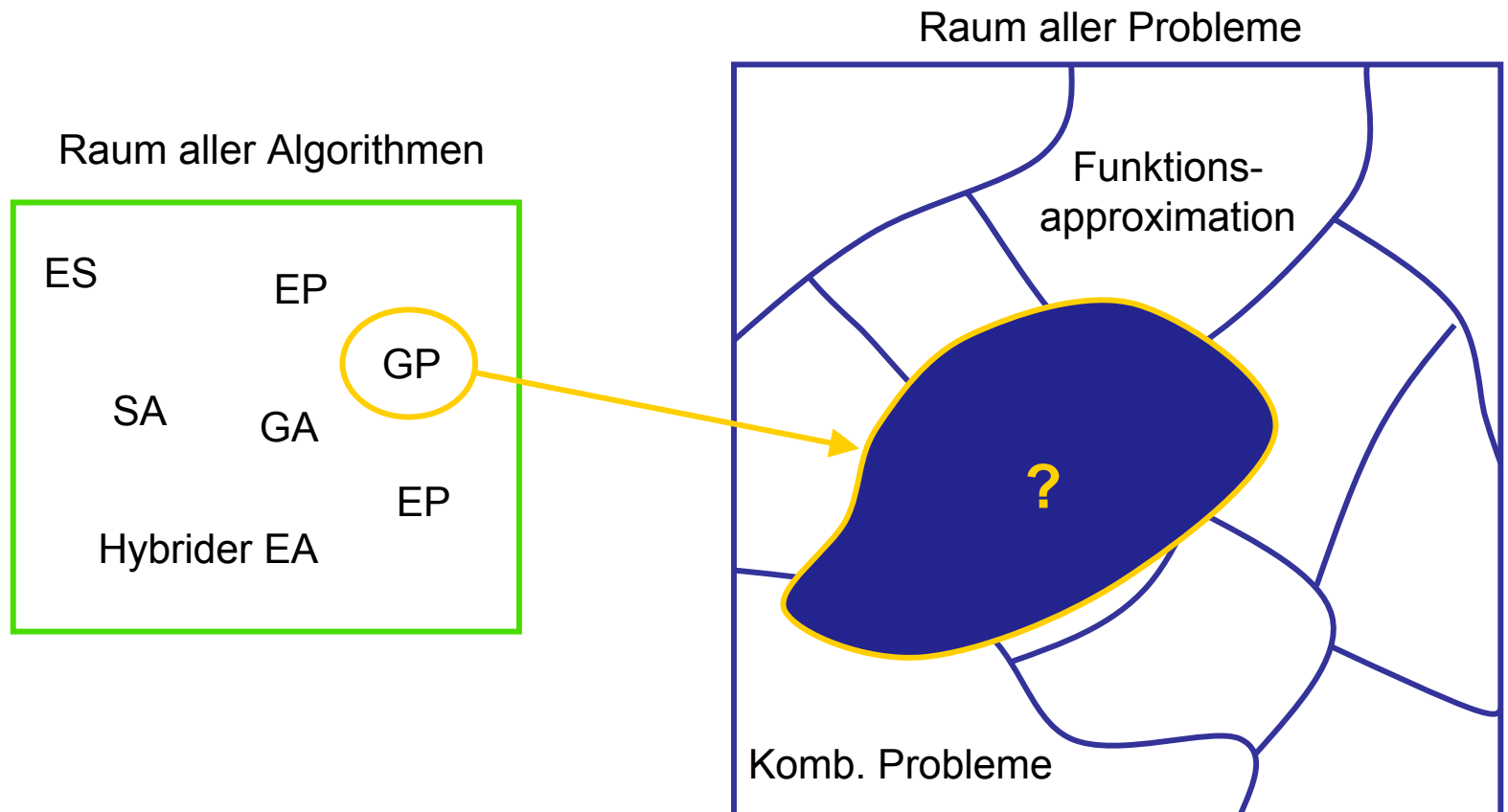
$$E[Q(Alg^*(F \setminus F', n))] < E[Q(Alg_i(F \setminus F', n))]$$

Also: Suche nach einem guten – dem besten – Algorithmus für eine Klasse von Problemen!

Konsequenzen aus dem NFL für einen Anwender



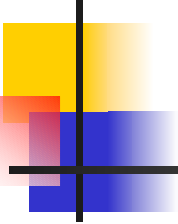
Konsequenzen aus dem NFL für die Wissenschaft





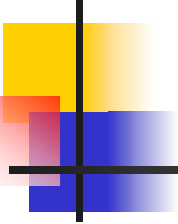
Abhängigkeiten der Parameter

- Parameter beeinflussen das Verhalten von EAs sehr stark
 - Auswahl der evolutionären Operatoren
 - Wahrscheinlichkeiten für evolutionäre Operatoren
 - Populationsgröße, Anzahl der Nachkommen
 - Auswahl der Selektionsmethoden
 - Bewertung der Individuen
- Ziel einer *guten* Parameterwahl
 - Erforschung neuer Regionen im Suchraum (**Exploration**)
 - Feinabstimmung des bereits durch die Suche eingegrenzten Gebiets (**Exploitation**)
- Parameter beeinflussen außerdem
 - Konvergenzgeschwindigkeit
 - Diversität in der Population
- Nur wenige (theoretische) Regeln zur Parameterwahl bekannt
 - idR experimentell für spez. Algorithmus und geg. Problemstellung
- Adaptive Algorithmen passen Parameter während des Evolutionsprozesses an



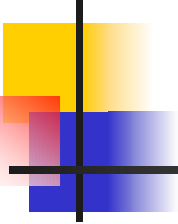
Gliederung

- 3.1 Einleitung: Universelle Anwendbarkeit eines EAs?
- 3.2 Selektionsmethoden
- 3.3 Evolutionäre Operatoren
- 3.4 Schema-Theorem und andere theoretische Resultate
- 3.5 Modelle für den Suchprozess



3.4 Schema-Theorem

- Schematheorie wurde für **Genetische Algorithmen** entwickelt
→ Kodierung der Individuen durch Bitstrings fester Länge
- **Idee:** Parallele Suche auf Hyperebenen im Suchraum
- Suchraum lässt sich in Äquivalenzklassen unterteilen, wenn man bestimmte Eigenschaften für Individuen definiert (auf genotypischer und phänotypischer Ebene)
- Jede Population stellt Menge von Stichproben der Äquivalenzklassen dar
- Ziel ist es Eigenschaften zu finden, die Individuen *positiv* beeinflussen → **Merkmale**



Merkmale und Formae

Definition:

Sei Ψ eine Menge von Merkmalen, so dass für jedes $\psi \in \Psi$ und zwei beliebige Individuen $A, B \in G$ gilt:

$A \sim_{\psi} B$, falls das Merkmal identisch ist, oder
 $\neg(A \sim_{\psi} B)$, falls das Merkmal nicht identisch ist.

Jede Äquivalenzklasse $[A]_{\psi} = \{B \mid A \sim_{\psi} B\}$ bezeichnet man als **Forma**.

- Die Anzahl der Formae, die durch ein Merkmal eingeführt wird, heißt **Genauigkeit** des Merkmals.
- Zwei Merkmale sind **verträglich**, wenn es ein Individuum gibt, das beide Merkmale erfüllt.
- **Beispiel** für ein Merkmal: Routen beim Handlungsreisenproblem, die nach Stadt 3 die Stadt 4 besuchen

Beispiel: Handlungsreisendenproblem

1. Möglichkeit:

- Betrachte alle Rundtouren mit Stadt i auf der i -ten Position
- Betrachte Merkmal $\psi = \{3\}$ $\longrightarrow A \sim_{\psi} B \Leftrightarrow A_3 = B_3$

$\longrightarrow$ Forma
$$[(1,2,3,4)]_{\psi} = \left\{ (1,2,3,4), (1,4,3,2), (2,1,3,4), \right. \\ \left. (2,4,3,1), (4,2,3,1), (4,1,3,2) \right\}$$

- **Aber:** Nicht absolute Position einer Stadt sondern Relation zu benachbarten Städten ist relevant!

Beispiel: Handlungsreisendenproblem

1. Möglichkeit:

- Betrachte alle Rundtouren mit Stadt i auf der i -ten Position
- Betrachte Merkmal $\psi = \{3\}$ $\longrightarrow A \sim_{\psi} B \Leftrightarrow A_3 = B_3$

$\longrightarrow$ Forma $[(1,2,3,4)]_{\psi} = \left\{ (1,2,3,4), (1,4,3,2), (2,1,3,4), \right. \\ \left. (2,4,3,1), (4,2,3,1), (4,1,3,2) \right\}$

- **Aber:** Nicht absolute Position einer Stadt sondern Relation zu benachbarten Städten ist relevant!

2. Möglichkeit:

- Betrachte alle Rundtouren, die nach Stadt i Stadt j besuchen:

$$\psi' = \{3\}$$

$$A \sim_{\psi'} B \Leftrightarrow \exists_{i,j \in \{1,\dots,l\}} (A_i = 3 \wedge B_j = 3 \wedge A_{(i \bmod l)+1} = B_{(j \bmod l)+1})$$

$\longrightarrow$ Forma $[(1,2,3,4)]_{\psi'} = \left\{ (1,2,3,4), (2,1,3,4), (2,3,4,1), (1,3,4,2), \right. \\ \left. (3,4,1,2), (3,4,2,1), (4,1,2,3), (4,2,1,3) \right\}$



Masken und Schemata

Annahme: $G = \mathbf{G}^l$ für feste Länge l

Merkmale fassen mehrere Positionen zusammen



Definition von Hyperebenen im Suchraum

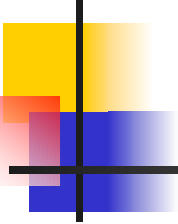
Beschränkung der Suche auf Merkmale mit *positivem* Einfluss auf die Fitness

Definition:

Für einen Genotyp der Form $G = \mathbf{G}^l$ wird die Menge aller genotypischen Masken als $\Psi = \text{Pot}(\{1, \dots, l\})$ (= Potenzmenge) definiert. Jede Maske ψ definiert eine Äquivalenzrelation auf G :

$$A \sim_{\psi} B :\Leftrightarrow \forall_{i \in \psi} A_i = B_i$$

Die Äquivalenzklassen von $\sim_{\psi}$ heißen **Schemata**.



Die **Ordnung** einer Maske $\text{ord}(\psi)$ ist die Anzahl der definierten Positionen: $\text{ord}(\psi) = \#\psi$

Die **Länge** einer Maske ist die maximale Entfernung zweier Indizes, d.h. $\delta(\psi) = \max\{|i - j| \mid i, j \in \psi\}$

Für ein Alphabet der Größe $\#G$ und eine Maske ψ gibt es $\#G^{\text{ord}(\psi)}$ verschiedene Schemata mit jeweils $\#G^{l - \text{ord}(\psi)}$ Elementen.

Schreibweise für Maske H : $H = \{H_1, \dots, H_n\} \in (G \cup \{*\})^l$

$$\text{mit } H_i = \begin{cases} A_i, & \text{falls } i \in \psi \\ *, & \text{sonst} \end{cases}$$

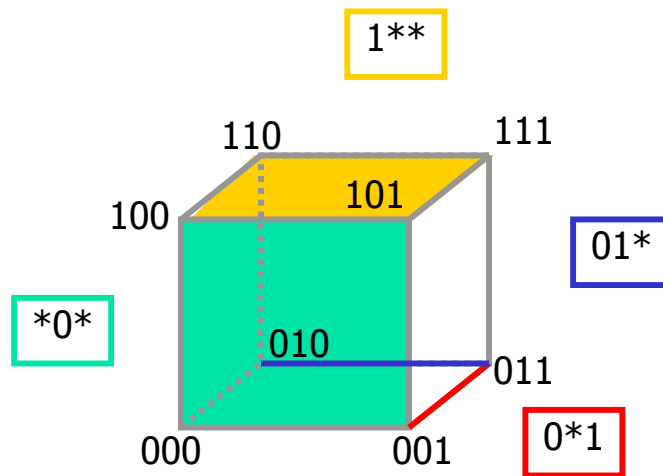
Beispiel: $G = \{0,1\}^6$, Maske ist $\psi = \{2,4,5,6\}$

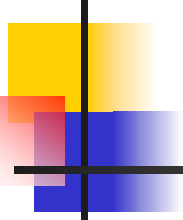
damit werden 16 Schemata mit jeweils 4 Individuen beschrieben

z.B.

$$[(*,0,*,0,1,0)]_{\psi} = \left\{ \begin{array}{l} (0,0,0,0,1,0), (0,0,1,0,1,0), \\ (1,0,0,0,1,0), (1,0,1,0,1,0) \end{array} \right\} \quad \begin{array}{l} \text{ord}(\psi) = 4 \\ \delta(\psi) = 4 \end{array}$$

Definition von Hyperebenen in B^3 :



- 
- Schema-Theorie befasst sich mit Aussagen bzgl. der Äquivalenzklassen (Formae)
-

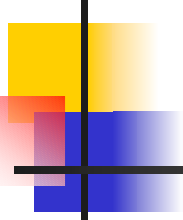
- Konkret:

Berechnung der Überlebenswahrscheinlichkeit von Formae aus einer Maske Δ unter Mutation und Rekombination (hier: Crossover) an einem Punkt

- Mutation M mit Mutationswkt. p_M pro Position
- pr_M : Wkt., dass Individuum mit Forma Δ nach der Mutation auch noch zu Δ gehört

Es gilt: $pr_M = (1 - p_M)^{\text{ord}(\Delta)}$

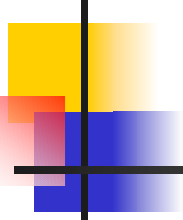
da p_M klein ist gilt $pr_M = 1 - p_M \text{ ord}(\Delta)$

- 
- Sei $\text{Prob}(\Delta, P(t))$ die Wahrscheinlichkeit, dass ein beliebiges Individuum aus Population $P(t)$ zu Form Δ gehört, also

$$\text{Prob}(\Delta, P(t)) = \frac{\#\{i \in P(t) \mid i \in \Delta\}}{\#P(t)}$$

- Sei p_c die Wahrscheinlichkeit eines Crossovers für Individuen A und B , d.h.
 - zufällige Auswahl einer Position $k \in \{1, \dots, l-1\}$
 - Erzeugung zweier neuer Individuen $(A_1, \dots, A_k, B_{k+1}, \dots, B_l)$ und $(B_1, \dots, B_k, A_{k+1}, \dots, A_l)$
- Für die Überlebenswkt. bei Crossover gilt:
 - Falls zwei Individuen aus Δ rekombiniert werden, so sind auch die Nachkommen in Δ
 - Falls zwei Individuen rekombiniert werden, von denen nur eines in Δ liegt, so ist kein Nachkommen in Δ , falls der Crossoverpunkt in die definierende Länge der Maske fällt

Damit gilt die Überlebenswahrscheinlichkeit:


$$\Pr_C = 1 - p_C \frac{\delta(\Delta)}{l-1} (1 - \text{Prob}(\Delta, P(t)))$$

Herleitung:

Überlegung, unter welchen Bedingungen ein Forma zerstört wird,
d.h. kein Nachkomme zur Äquivalenzklasse gehört

Voraussetzung: Ein Elternteil gehört zu Δ

- Mit Wahrscheinlichkeit $p_C \frac{\delta(\Delta)}{l-1}$

findet ein Crossover statt, bei dem der Schnittpunkt in die definierende Länge der Maske fällt

- Damit kein Nachkomme zu Δ gehört, muss zusätzlich der Partner nicht zu Δ gehören

➡ Wahrscheinlichkeit $1 - \text{Prob}(\Delta, P(t))$

Vereinfachte Berechnung:

$$\Pr_C \approx 1 - p_C \frac{\delta(\Delta)}{l-1}$$



Analyse des Selektionsoperators:

→ proportionale Selektion

$$\overline{F(t)} = \frac{1}{p} \sum_{i=1}^p A^{(i)} \cdot F \quad \text{und} \quad \overline{F_{\Delta}} = \frac{1}{p \text{Prob}(\Delta, P(t))} \sum_{i \in \Delta} A^{(i)} \cdot F$$

Auswahlwkt. eines Individuums aus Δ

$$pr_s = \frac{\text{Prob}(\Delta, P(t)) \cdot \overline{F_{\Delta}}}{\overline{F(t)}}$$

Damit gilt für die nächste Generation $t+1$:

$$\text{Prob}(\Delta, P(t+1)) \geq pr_M pr_C pr_s$$

→ rasche Vermehrung von Schemata mit

- überdurchschnittlicher Fitness
- kleiner definierender Länge
- geringer Ordnung

→ **Building Blocks**



Bisherige Aussagen betrachten Übergang von $P(t)$ nach $P(t+1)$



Verallgemeinerung von $P(0)$ nach $P(t)$ (für beliebiges t)

Sei Δ ein Forma überdurchschnittlicher Güte, d.h.

$$\overline{F_{\Delta}} = \overline{F(t)} + c \overline{F(t)}, c > 0$$

und kleiner definierender Länge,

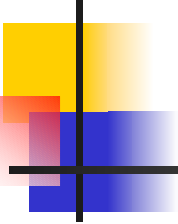
dann gilt (bei Annahme konstanter durchschnittlicher Güte)

$$\text{Prob}(\Delta, P(t+1)) \approx \text{Prob}(\Delta, P(0))(1+c)^t$$



Baustein-Hypothese:

gute Bausteine vermehren sich
exponentiell in einer Population



Beispiel

- Einsetzen der Wahrscheinlichkeiten
- Damit gilt für die nächste Generation $t+1$:

$$\text{Prob}(\Delta, P(t+1)) \geq (1 - p_M)^{\text{ord}(\Delta)} \left(1 - p_C \frac{\delta(\Delta)}{l-1} \right) \frac{\text{Prob}(\Delta, P(t)) \cdot \overline{F_\Delta}}{\overline{F(t)}}$$

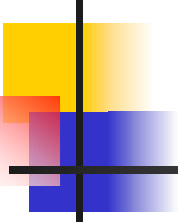
Beispiel

Population P(t) und Schema H=****1111****

Individuum	Güte	Individuum	Güte
111100000000	4	111101011010	4
000011110000	4	111111110011	8
000000001111	4	111110001111	8
100011111111	8	111001110111	0

$$\begin{aligned}
 \text{Prob}(\Delta, t+1) &\geq \frac{1}{2} \left(1 - \frac{1}{12}\right)^4 \left(1 - 0.8 \frac{3}{11} \left(1 - \frac{1}{2}\right)\right) \\
 &= \frac{1}{2} \left(\frac{11}{12}\right)^4 \frac{98}{100} = 0.3142
 \end{aligned}$$

Da $\text{Prob}(\Delta, t) = \frac{3}{8} = 0.375$, nimmt die Häufigkeit des Schemas H trotz guter Fitness ab (Länge und Ordnung von H sind zu groß)



Kritik an der Schema-Theorie

- Theorie verallgemeinert Resultate für einen Schritt auf eine Folge von Schritten (→ exponentielles Güterwachstum), aber Änderung der durchschnittlichen Güte wird nicht berücksichtigt
- Populationen sind in der Praxis im Vergleich zum Suchraum sehr klein → Wahrscheinlichkeitsaussagen gelten nicht zwangsläufig
- Sich stark vermehrende Schemata müssen nicht zwangsläufig gut für die Optimierung sein → gute können von Beginn an fehlen
- Schemata mit großer definierender Länge werden mit hoher Wkt. zerstört



Entwurfsprinzipien auf Basis der Schema-Theorie

Ziel: überproportionales Wachstum hochwertiger Formae

- **Minimale Redundanz:** Jede Komponente des Genotyps sollte Information liefern

→ für $A, B \in G$, mit $A \neq B$ mit $dec(A) = dec(B)$
soll auch $B \in [A]_{\psi}$ gelten

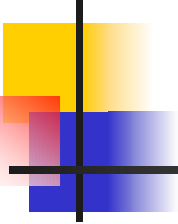
- **Ähnlichkeit der Formae:** Masken sind so zu wählen, dass Individuen mit ähnlicher Fitness und ähnlichem Phänotyp zusammengefasst werden
- **Abschluss der Formae gegen Schnitt:** einzelne Formae sollten kombinierbar sein, um Formae mit höherer Genauigkeit zu erzeugen
- **Erreichbarkeit durch Mutation:** Mutationsoperator muss sicher stellen, dass jeder Punkt im Suchraum erreichbar ist



Entwurfsprinzipien für die Rekombination

- **Verträglichkeit mit Formae:** Nachkommen zweier Individuen aus einem Forma müssen auch zu diesem Forma gehören
- **Übertragung von Genen:** Formae minimaler Genauigkeit, die von Kindern erfüllt werden, sollen von mindestens einem Elternteil ausgehen
- **Verschmelzungseigenschaft:** zwei Individuen, die kompatible Formae erfüllen, lassen sich so rekombinieren, dass ein Kind beide Formae erfüllt

Idee hier: Mutation zur Erforschung
Rekombination zur Feinabstimmung



Gliederung

- 3.1 Einleitung: Universelle Anwendbarkeit eines EAs?
- 3.2 Selektionsmethoden
- 3.3 Evolutionäre Operatoren
- 3.4 Schema-Theorem und andere theoretische Resultate
- 3.5 Modelle für den Suchprozess



3.5 Modelle für den Suchprozess

Ziel:

- theoretische Untersuchung des gesamten Ablaufs
- Aussagen zu folgenden Problemen treffen
 - Konvergenz des Verfahrens gegen ein globales Optimum (Bedingung an Parameter und Funktion)
 - Wahrscheinlichkeit nach endlich vielen Schritten *in der Nähe* des globalen Optimums zu sein
 - Bestimmung der Konvergenzgeschwindigkeit
 - Optimierung der Parametereinstellung
 - Fundierte Herleitung von Heuristiken bei EA-Design
- Zur Analyse ist ein Modell des Algorithmus notwendig



Verwendete Modelle:

- Schema Theorie
- Markov-Ketten
- Statistische Physik
- ...

➡ Alle Modelle sind bisher nur für relativ einfache Probleme und Algorithmen einsetzbar