Evolutionäre Algorithmen

Vorlesung 4

Evolutionäre Operatoren

Evolutionäre Operatoren

- ▷ Gegenstücke zur Selektion
- ▷ Erzeugung neuer Lösungskandidaten
- ▷ bestimmen mögliche Richtungen, die eine Suche nehmen kann
- ▷ unärer Operator: Mutation
- ▷ k-ärer Operator: Rekombination

Mutation: zwei Rollen

- ▷ kleine Veränderung am Phänotyp: Feinabstimmung bzw. *exploitation*
- ▷ große Veränderung am Phänotyp: Erforschung bzw. *exploration*
- ▷ Feinabstimmung ist kritischer als Erforschung

  $\Rightarrow$ meist: kleine Veränderung (am Genotyp!)

Mutation: lokale Suche

- ▷ Mutation nimmt kleine Veränderung vor
- ▷ Selektion bevorzugt bessere Individuen
- ▷ Wechselspiel zwischen Mutation und Selektion erzeugt eine lokale Suche

Mutation: Nachbarschaftsgraph

- endlicher genotypischer Suchraum $\mathcal{G}$

Definition: Nachbarschaftsgraph

- Mutationsoperator $M^{\xi} : \mathcal{G} \to \mathcal{G}$
- Nachbarschaftsgraph $G = (V, E)$ mit
- Knoten $V = \mathcal{G}$ und
- Kanten

$$E = \left\{ (v_1, v_2) \,|\, \exists \xi \in \Xi : M^{\xi}(v_1) = v_2 \right\} \subset V \times V$$

Mutation: Nachbarschaftsgraph

- Was lässt sich an Nachbarschaftsgraphen ablesen?

- veranschaulicht: Erreichbarkeit der Lösungskandidaten
- minimaler Zeitbedarf für Veränderungen an Individuen
- lokale Optima

Mutation: Lokales Optimum

Definition: Lokales Optimum

- Mutationsoperator $M^{\xi} : \mathcal{G} \to \mathcal{G}$
- induzierte Bewertungsfunktion $f : \mathcal{G} \to \mathbb{R}$
- Nachbarschaftsgraph $G = (V, E)$ zu M
- Dann heißt $v \in V$ lokales Optimum falls
  für alle $(v, v') \in E$ gilt: $f(v) \succeq f(v')$

Mutation: Bedeutung der Kodierung

- gilt $\mathcal{G} \neq \mathcal{S}$
- dann bestimmen Kodierung und Mutation die Anzahl und Position der lokalen Optima
- es können mehr lokale Optima entstehen als mit der natürlichen Nachbarschaftsbeziehung
- oft vorteilhaft: Einbettung der phänotypischen Nachbarschaft in die mutationsdefinierte, genotypische Nachbarschaft

Mutation: Beispiel für Kodierungen

- ▷ Kodierung von $[u, v]$ mit l Bits
- ▷ standardbinäre Kodierung

  $$decode_{stdbin}(A) \;=\; u + \frac{v-u}{2^l-1}\left(\sum_{j=1}^{l} A_{l-j} 2^{j-1}\right)$$

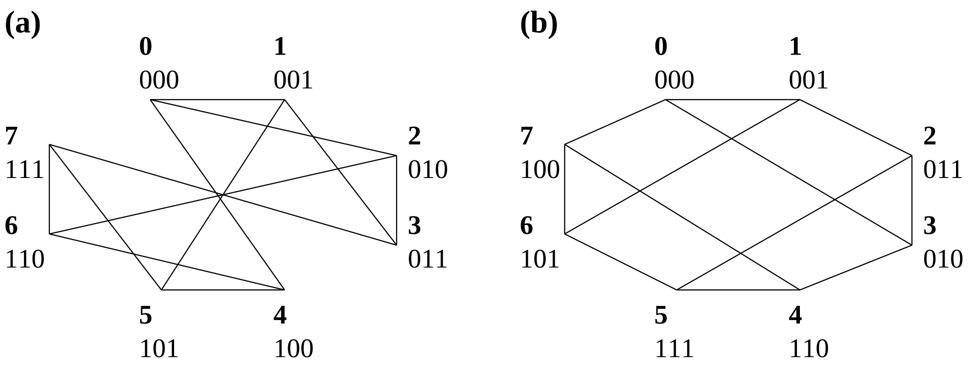
- ▷ Gray-Kodierung

  $$decode_{Gray}(A) \;=\; decode_{stdbin}(Gray2stdbin(A))$$

  mit $Gray2stdbin(A_1 \cdots A_l) = (B_1 \cdots B_l)$ gegeben

  durch $B_i \;=\; \bigoplus_{j=1}^{i} A_j$

Mutation: Beispiel für Kodierungen



Mutation: Exploration

- ▷ Allerdings: auch nicht-lokale Veränderungen sind wichtig
- ▷ erhalten die Diversität im Suchraum
- ▷ soll gewährleisten, dass alle Punkte im Suchraum (schnell) erreicht werden können

Ausschließlich lokale Mutation

- ▷ Mutation und Selektion ergeben lokale Suche
- ▷ sehr lokale Mutation kann in lokalen Optima gefangen sein
- ▷ Dann: Erreichbarkeit aller Punkte geht verloren

  $\Rightarrow$ Mutation muss auch immer diversitätserhaltend sein
- ▷ Andernfalls: anderer explorativer Operator ist notwendig

Rekombination: Kombination

- hat meist keine eigene „Kreativität“
- kombiniert i.d.R. Informationen der Eltern
- untermauert durch „Schema/Forma-Theorie“: parallele Suche in den Hyperebenen der Eigenschaften
- meist genotypische, selten phänotypische Eigenschaften

Rekombination: Rolle der Diversität

- Rekombination benötigt Diversität
  ⇒ nur dann findet effektiv durch neue Impulse eine Suche in den Hyperebenen statt
- fehlt die Diversität: viele Individuen sind sehr ähnlich zueinander
  ⇒ ein kontrahierender Operator kann große Ausreisser durch die Mutation statistisch herausmitteln (genetisches Reparieren)

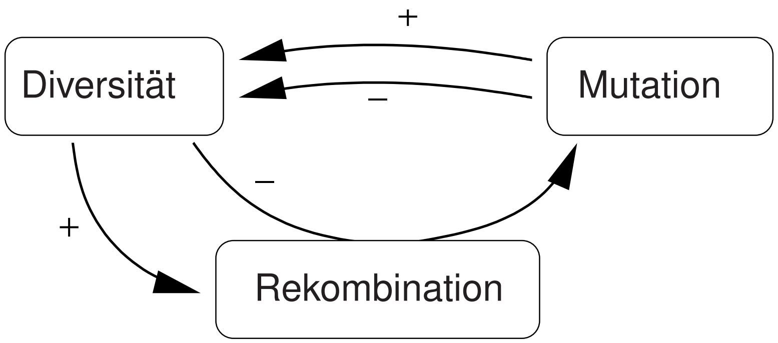
Rekombination: Extrapolation

- für eine mehr gerichtete Suche durch die Rekombination sind die Gütewerte zu benutzen
- daraus läßt sich eine Richtung ableiten, in der bessere Gütewerte erwartet werden
  ⇒ Extrapolation

Wechselwirkungen zwischen Operatoren

- große Mutationen ⇒ große Diversität
  große Diversität ⇒ effektive Rekombination durch große Menge an unterschiedlichen Samples
  D.h. beide Operatoren agieren eher explorativ
- Mutation klein ⇒ kleine Diversität
  kleine Diversität ⇒ Rekombination wirkt glättend auf Mutation
  D.h. beide Operatoren bestärken lokale Suche (Exploitation)

Regelkreise



Folgerungen aus Regelkreisen

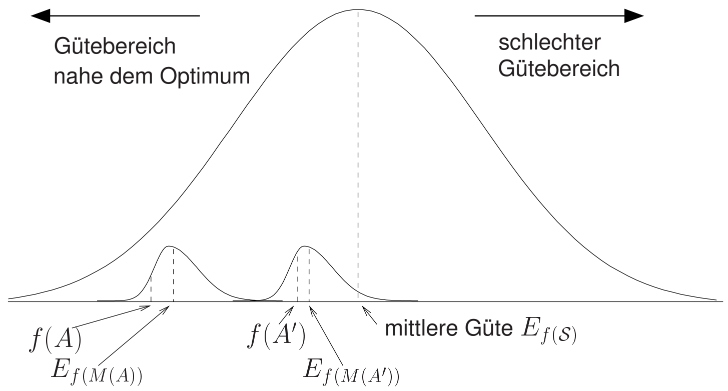
- ▷ zentrale Rolle der Diversität
- ▷ für große Exploration (z.B. zu Beginn der Suche)

  $\Rightarrow$ große Mutationen
- ▷ für Exploitation (z.B. gegen Ende der Suche)

  $\Rightarrow$ kleine Mutationen
- ▷ Faktoren wie Populationsgröße, Selektion, etc. fehlen noch!

Einfluss des Stands der Suche

- ▷ Betrachten: Gütehistogramme



Idealisiertes Szenario

- ▷ Annahme: verschiedene Operatoren verhalten sich ähnlich

  $\Rightarrow$ unterscheiden sich nur in Bezug auf ihre Gütevarianz
- ▷ stärkere Verschiebung des Erwartungswerts der Güte der Nachkommen von M falls
  - Ausgangspunkt näher am Optimum
  - größere Gütevarianz des Operators
- ▷ $E_{f(M(A)} - f(A) = \dfrac{\sigma^2_{f(M(A))}}{\sigma^2_{f(\mathcal{S})}} \Big(E_{f(\mathcal{S})} - f(A) \Big)$

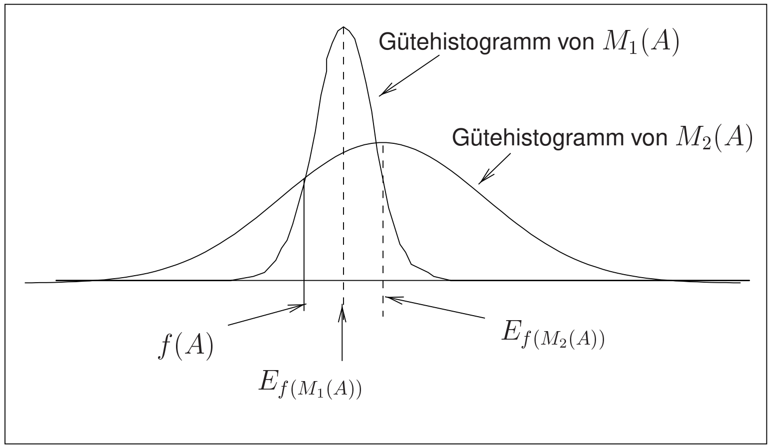
Vergleich von Operatoren

- ▷ (Multi-)Menge aller möglichen Nachkommen $T_{M,A}$
- ▷ (Multi-)Menge aller besseren Nachkommen $\hat{T}_{M,A}$
- ▷ Erfolgswahrscheinlichkeit: $Q_{M,A} = \frac{|\hat{T}_{M,A}|}{|T_{M,A}|}$
- ▷ erwartete Verbesserung:

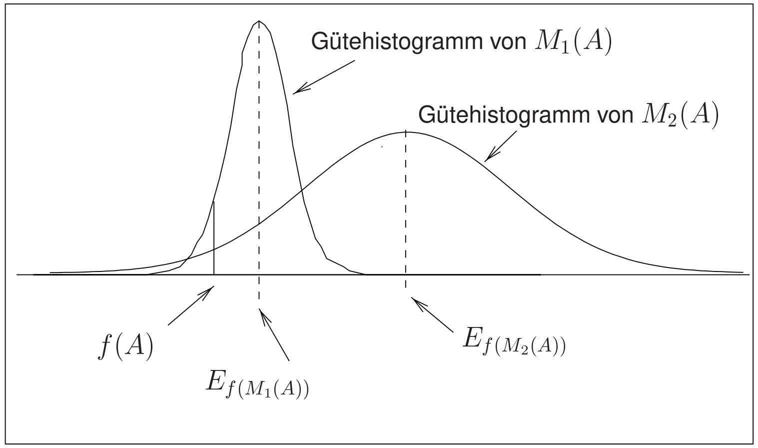
  $P_{M,A} = \frac{1}{|\hat{T}_{M,A}|} \sum_{B \in \hat{T}_{M,A}} |f(B) - f(A)|$
- ▷ Kombination aus beidem:

$$R_{M,A} = Q_{M,A} P_{M,A} = \frac{1}{|T_{M,A}|} \sum_{B \in \hat{T}_{M,A}} |f(B) - f(A)|$$

Nahe der mittleren Güte von $\mathcal{S}$



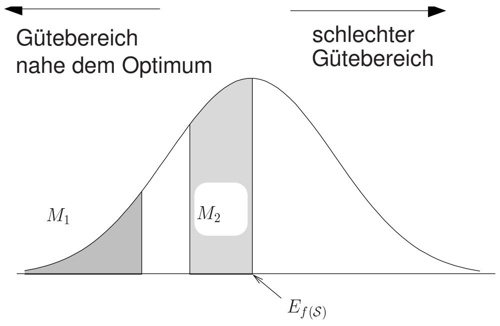
Nahe der besten Güte



Einfluss des Stands der Suche

- ▷ für beide Operatoren existiert jeweils ein

  Gütebereich, auf dem sie dem anderen überlegen sind.



- ▷ lässt sich unter bestimmten Bedingungen theoretisch zeigen

Folgerungen

- für eine schnelle Optimierung gilt:
- gütebezogenes Verhalten der Mutation hat sich im Verlauf der Optimierung zu verändern
- erneut: Wechsel von Exploration zur Exploitation ist notwendig
- Parameter des Algorithmus müssen während des Suchverlaufs verändert werden

Vordefinierte Parameteranpassung

- vor der Optimierung wird ein Plan aufgestellt, wie die Parameter verändert werden
- **Nachteil:** unterschiedliche Probleme und Probleminstanzen benötigen unterschiedliche Anpassung
- kann zu verfrühter Konvergenz auf schlechtem Lösungskandidaten führen
- kann zu sehr langsamer Konvergenzgeschwindigkeit führen

Adaptation

- Informationen aus dem Suchprozess werden genutzt, um die Parameter anzupassen
- global für die gesamte Population oder für einzelne Individuen
- **Vorteil:** tatsächliche Berücksichtigung der aktuellen Optimierung ist möglich
- **Nachteil:** oft sind Indizien und Formeln zur Anpassung der Parameter nur schwierig zu finden

Selbst-Adaptation

- Parameter werden in jedem Individuum kodiert
- sie unterliegen ebenfalls der Selektion und haben sich geeignet herauszubilden
- **Vorteil:** Parameteranpassungen sind möglich, die nicht durch allgemeine Regeln spezifiziert werden können
- **Nachteil:** Man hat keinen Einfluss mehr auf das Verhalten der Parameter

Theorie für erfolgreiche Adaptation

- ▷ **gesucht:** Theorie zur Adaptation und Selbstadaptation
- ▷ Es ist eine offene Frage, wann Regeln zur Adaptation wirklich erfolgreich sind
- ▷ ebenso offen: wann lassen sich Selbst-Adaptationen in die Irre leiten?