UNIKASSEL VERSITÄT

SOFTWARE ENGINEERING
UNIVERSITÄT KASSEL

Multikriterielle evolutionäre Algorithmen: Grundlagen

Jens Kosiol

(teilweise basierend auf Folien von Eiben/Smith)

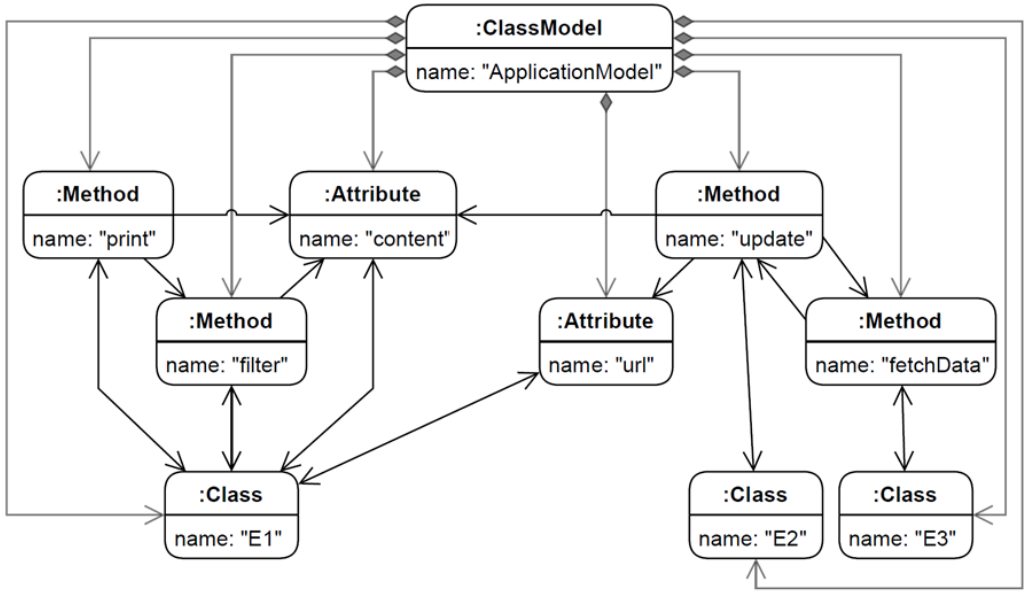
Wofür multikriterielle Optimierung?

- Multikriterielle Optimierung (multi-objective optimization) bezeichnet das gleichzeitige Optimieren in Bezug auf mehrere Optimierungsziele (Fitnessfunktionen).
  - Weitere Namen: Pareto-Optimierung, Mehrzieloptimierung, Vektoroptimierung (Pareto optimization, vector optimization)
  - In der Literatur wird noch genauer zwischen multi-objective (2–3 Optimierungsziele) und many-objective (mehr als 3 Optimierungsziele) unterschieden.
- In den meisten praktisch relevanten Optimierungsproblemen gibt es mehr als ein Optimierungsziel.
  - Ein häufiges Muster: Maximieren der Qualität und minimieren der Kosten

Beispiel: Class Responsibility Assignment (Wdh.)

Kohäsionsrate: $\sum_{c \in C}\left(\frac{\mathrm{MAI}(c,c)}{|\mathrm{M}(c)| \times |\mathrm{A}(c)|} + \frac{\mathrm{MMI}(c,c)}{|\mathrm{M}(c)| \times |\mathrm{M}(c)-1|}\right)$

Kopplungsrate: $\sum_{\substack{c_i, c_j \in C \\ c_i \neq c_j}}\left(\frac{\mathrm{MAI}(c_i,c_j)}{|\mathrm{M}(c_i)| \times |\mathrm{A}(c_j)|} + \frac{\mathrm{MMI}(c_i,c_j)}{|\mathrm{M}(c_i)| \times |\mathrm{M}(c_j)|}\right)$



Hierbei gilt:

$$\mathrm{MMI}(c_i, c_j) = \sum_{\substack{m_i \in \mathrm{M}(c_i) \\ m_j \in \mathrm{M}(c_j)}} \mathrm{DMM}(m_i, m_j)$$

$$\mathrm{MAI}(m_i, a_j) = \sum_{\substack{m_i \in \mathrm{M}(c_i) \\ a_j \in \mathrm{A}(c_j)}} \mathrm{DMA}(m_i, a_j)$$

$\mathrm{A}(c)$: Die Attribute einer Klasse c

$\mathrm{M}(c)$: Die Methoden einer Klasse c

$$\mathrm{DMA}(m_i, a_j) = \begin{cases} 1, \text{falls Abhängigkeit zw. } m_i \text{und } a_j \\ 0, \text{sonst} \end{cases}$$

$$\mathrm{DMM}(m_i, m_j) = \begin{cases} 1, \text{falls Abhängigkeit zw. } m_i \text{und } m_j \\ 0, \text{sonst} \end{cases}$$

Klassischer Ansatz: Skalarisierung

- Bei der Skalarisierung wird die Menge der Fitnessfunktionen per Aggregation zusammengefasst.
- Einfachste Methode: Lineare Gewichtung. Sind Fitnessfunktionen $f_1, \dots, f_n$ zu maximieren (minimieren), wird die Funktion

$$f = \sum_{i=1}^{n} w_i f_i$$

  maximiert (minimiert), wobei durch die Gewichte w_i der Einfluss der einzelnen Optimierungsziele gesteuert werden kann.
- Schwierigkeiten:
  - Es ist schwer, a priori passende Gewichte zu bestimmen.
  - Häufig will man lieber aus einer Menge unterschiedlicher Kompromisse auswählen können.

Pareto-Optimalität

Intuition: Sind verschiedene Optimierungsziele gegeben, so ist eine Pareto-optimale Lösung (nondominated solution), eine Lösung, bei der man für jede Verbesserung im Hinblick auf ein Optimierungsziel eine Verschlechterung im Hinblick auf (mindestens) ein anderes Optimierungsziel in Kauf nehmen muss.

Formale Definition:
Sind Fitnessfunktionen $f_1, \dots, f_n$ gegeben, so (pareto)dominiert eine Lösung x eine Lösung y, falls

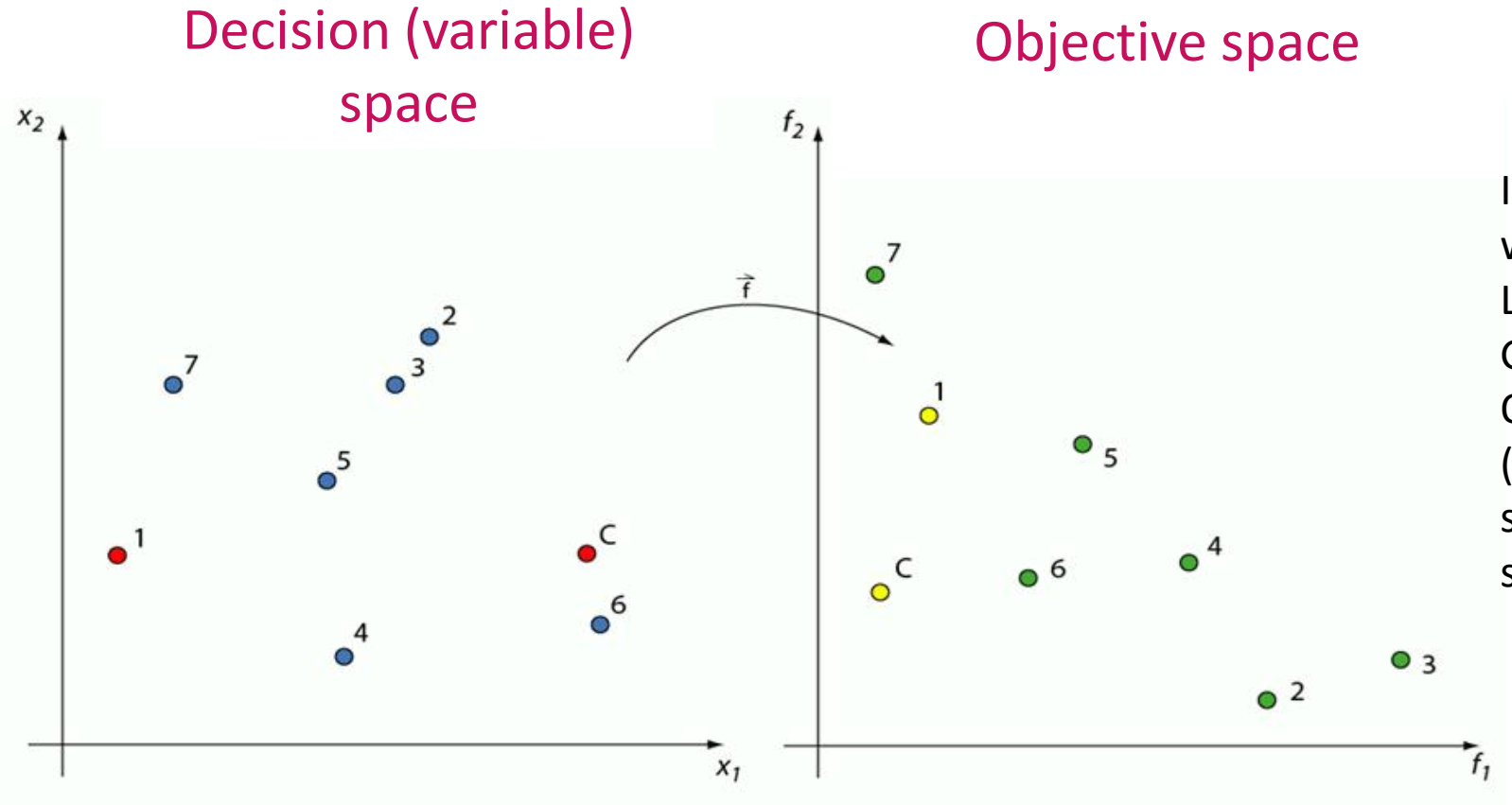
1. $f_i(x) \geq f_i(y)$ für alle $1 \leq i \leq n$ und
2. $f_i(x) > f_i(y)$ für (mindestens) ein $1 \leq i \leq n$

(für Maximierungsprobleme; für Minimierungsprobleme $\leq$ und $<$).

Eine Lösung x heißt Pareto-optimal (Englisch auch häufig: nondominated), falls es keine Lösung y gibt, die x paretodominiert. Die Pareto-Menge ist die Menge aller Pareto-optimalen Lösungen.

Decision und objective space

Der decision space ist der Suchraum, der die Lösungen enthält. Seine Form hängt von der gewählten Kodierung ab.

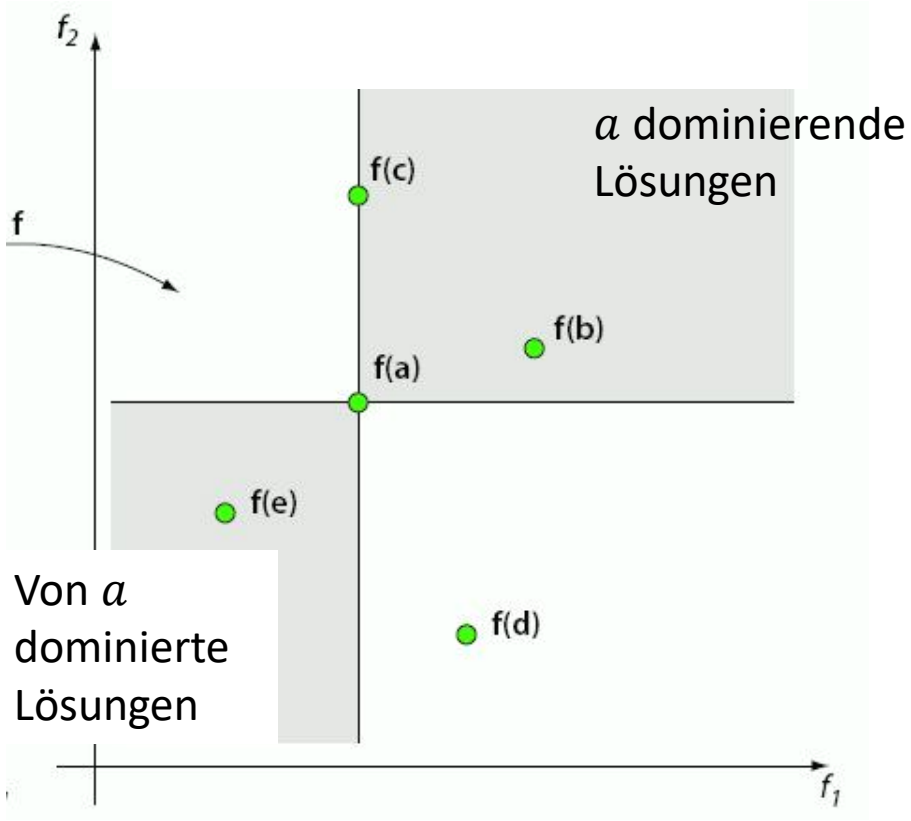


Im objective space wird die Fitness der Lösungen verglichen. Gibt es n Optimierungsziele (Fitnessfunktionen), so ist der objective space der $\mathbb{R}^n$.

Pareto-Front: Bild der Pareto-Menge im objective space
Bei Maximierungsproblem hier: $f(7), f(1), f(5), f(4), f(3)$

Beispiel: Vergleich von Lösungen

Objective space



Sind f_1 und f_2 zu maximieren, wie verhält sich die Qualität der Lösungen b, c, d, e im Vergleich zu der von a?

- a ist besser als e
- a ist schlechter als b und c
- a und d sind nicht vergleichbar

Ziele multikriterieller Optimierung

Ziele multikriterieller Optimierung:

- Eine Menge von Lösungen finden, die sich nicht gegenseitig dominieren (approximation set)
- Die Lösungen in dem berechneten approximation set sollen:
  - Möglichst nah an der Pareto-Front liegen (Konvergenz)
  - Sich untereinander möglichst unterscheiden (Diversität)

Warum evolutionäre Algorithmen?

Evolutionäre Algorithmen sind gut geeignet, multikriterielle Optimierungsprobleme zu lösen:

- Sie sind natürlicherweise populationsbasiert und daher geeignet, eine (diverse) Lösungs*menge* zu berechnen statt einer einzelnen Lösung.
- Evolutionäre Algorithmen sind unabhängig von der Form der Pareto-Front einsetzbar.
- Die verschiedenen Optimierungsziele können, müssen aber nicht a priori gewichtet werden.

Qualitätsmaße für multikriterielle Optimierung

Wofür Qualitätsmaße (Engl.: performance oder quality indicators)?

- Beobachtung zur Laufzeit zur
  - Steuerung des Algorithmus
  - Entscheidung über Termination
- Auswertung nach Termination zum Vergleich verschiedener Läufe, Parameterwahlen, Algorithmen

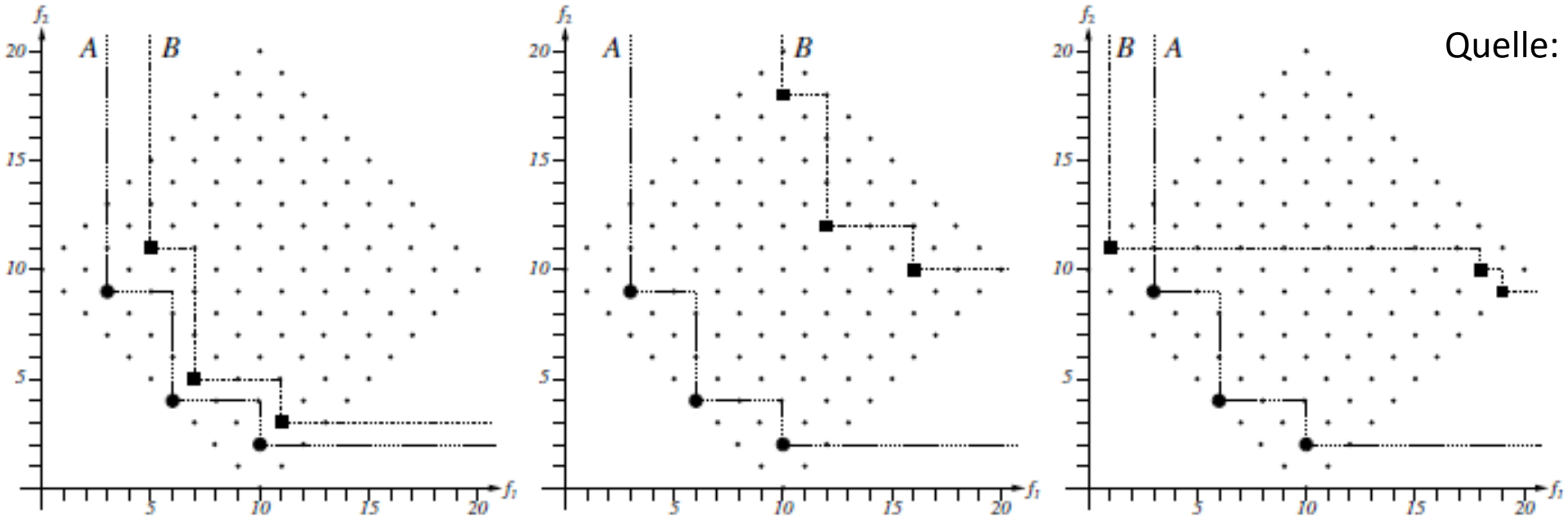
Fitnessvergleich von Mengen

Gegeben einen Suchraum (decision space) S und Fitnessfunktionen $f = (f_1, \dots, f_n)$ mit $f_i: S \to \mathbb{R}$, so können wir die Fitness von approximation sets/Populationen wie folgt vergleichen:

- Das Bild einer Population P im $\mathbb{R}^n$ (objective space) ist
$$f(P) = \{ z \in \mathbb{R}^n \mid \exists x \in P: z = f(x) \}.$$
- Für Populationen P_1, P_2 setzen wir (für Minimierungsprobleme)
$$P_1 \leq P_2 \Leftrightarrow \forall y \in P_2 \exists x \in P_1: x \leq y.$$
Wir schreiben $P_1 < P_2$ falls gilt $P_1 \leq P_2$ aber nicht $P_2 \leq P_1$.

- Wir erhalten drei mögliche Situationen:
  - Eine Population ist besser: $P_1 < P_2$ oder $P_2 < P_1$
  - P_1 und P_2 sind bezüglich $<$ gleich gut: Es gilt $P_1 \leq P_2$ und $P_2 \leq P_1$.
  - P_1 und P_2 sind durch $<$ nicht vergleichbar: Es gilt weder $P_1 \leq P_2$ noch $P_2 \leq P_1$.

Beispiel Fitnessvergleich Mengen



Quelle: [ZKT08]

Drei Vergleiche von Bildern von approximation sets im $\mathbb{R}^2$ (objective space):

- A dominiert B in den beiden ersten Abbildungen (Minimierungsproblem), in der zweiten aber deutlicher (ohne dass es für $\leq$ einen Unterschied macht).
- In der dritten Abbildung sind A und B nicht vergleichbar per $\leq$, in vielen Anwendungsfällen wird aber A vorzuziehen sein.

Zusätzliche Qualitätsmaße

Zusätzliche Qualitätsmaße (Engl.: quality oder performance indicators) für approximation sets sollen einen detaillierteren Vergleich ermöglichen, indem sie zusätzliche Eigenschaften wie Abdeckung des Suchraums, Diversität der Lösungen, Nutzerpräferenzen, … berücksichtigen.

Ein (einstelliges) Qualitätsmaß I ist eine Funktion $I: \Psi \to \mathbb{R}$, die jedem approximation set eine reelle Zahl zuordnet (Ψ steht für die Menge aller approximation sets). Typischerweise gilt: Gilt $I(P_1) > I(P_2)$, so ist P_1 der Menge P_2 vorzuziehen.

Eigenschaften von Qualitätsmaßen

Wünschenswerte Eigenschaften eines Qualitätsmaßes I:

- (Strenge) Monotonie: I sollte die von der Pareto-Optimalität induzierte Ordnung respektieren. Gilt also $P_1 \leq (<) P_2$, so sollte auch $I(P_1) \geq (>) I(P_2)$ gelten.
- Skalierungsinvarianz: Skalieren der Fitnessfunktionen sollte keinen Einfluss auf I haben.
- Komplexität: I sollte effizient berechnet werden können.
- Parameterunabhängigkeit: I sollte möglichst unabhängig von weiteren Parametern sein.

Beispiele für Qualitätsmaße

Einige Beispiele für Qualitätsmaße:

- Äußerer Durchmesser (outer diameter): Misst maximalen Fitnessunterschied, der für ein Optimierungsziel erreichbar ist
- Anteil an der Pareto-Front
- Kardinalität des approximation sets
- Hypervolumen (hypervolume): Misst Größe des dominierten Bereichs im objective space
- Vollständigkeit: Bestimmt Wahrscheinlichkeit eines beliebigen Punktes im Suchraum von einem Punkt aus dem approximation set dominiert zu werden
- Maße der Gleichförmigkeit: Misst Diversität eines approximation set z.B. durch Angabe der Standardabweichung der Distanz zum nächsten Nachbarn.

Outer diameter

Definition:

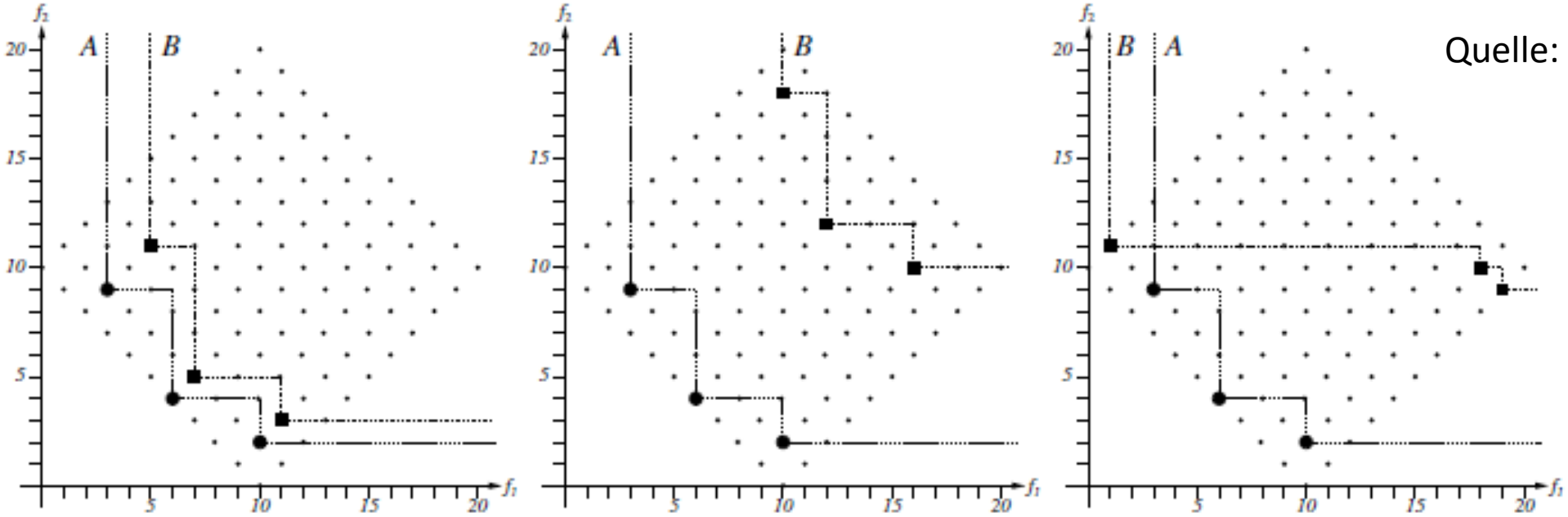
Gegeben Fitnessfunktionen $(f_1, \dots, f_n)$ und ein approximation set A berechnet sich der outer diameter als

$$I_{\mathrm{OD}}(A) = \max_{1 \le i \le n} w_i \left(\left(\max_{x \in A} f_i(x) \right) - \left(\min_{x \in A} f_i(x) \right) \right).$$

Eigenschaften:

- Weder monoton noch invariant gegenüber Skalierung der Fitnessfunktionen
- Leicht zu berechnen (linear in Größe des approximation set)
- Unabhängig von weiteren Parametern (durch Gewichte w_i können Präferenzen angegeben werden, aber sie können auch alle auf 1 gesetzt werden)

Illustration outer diameter



Quelle: [ZKT08]

Wie würde sich hier jeweils I_{OD} berechnen?

Anteil an der Pareto-Front

Definition:

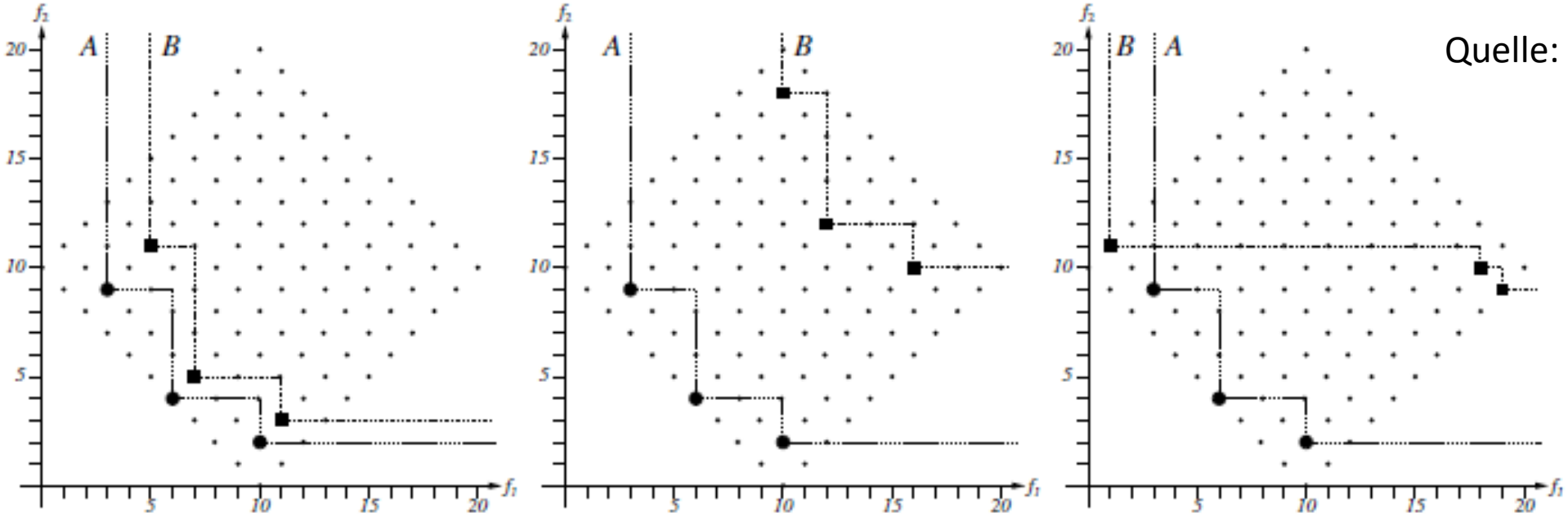
Gegeben (zu minimierende) Fitnessfunktionen $f = (f_1, \dots, f_n)$, die Pareto-Front $F \subseteq \mathbb{R}^n$ und ein approximation set A, berechnet sich der Anteil an der Pareto-Front als

$$I_{\mathrm{PF}}(A) = \frac{\{z \in F \mid \exists x \in A: f(x) \leq z\}}{|F|}$$

Eigenschaften:

- Monoton, aber nicht streng monoton
- Invariant gegenüber Skalierung der Fitnessfunktionen
- Leicht zu berechnen
- Nur einsetzbar, wenn Pareto-Front bekannt und endlich

Illustration Anteil an der Pareto-Front



Quelle: [ZKT08]

Wie würde sich hier jeweils I_{PF} berechnen, wenn das diskrete gepunktete Quadrat das Bild der validen Lösungen im objective space darstellt?

Definition Hypervolumen

Definition:

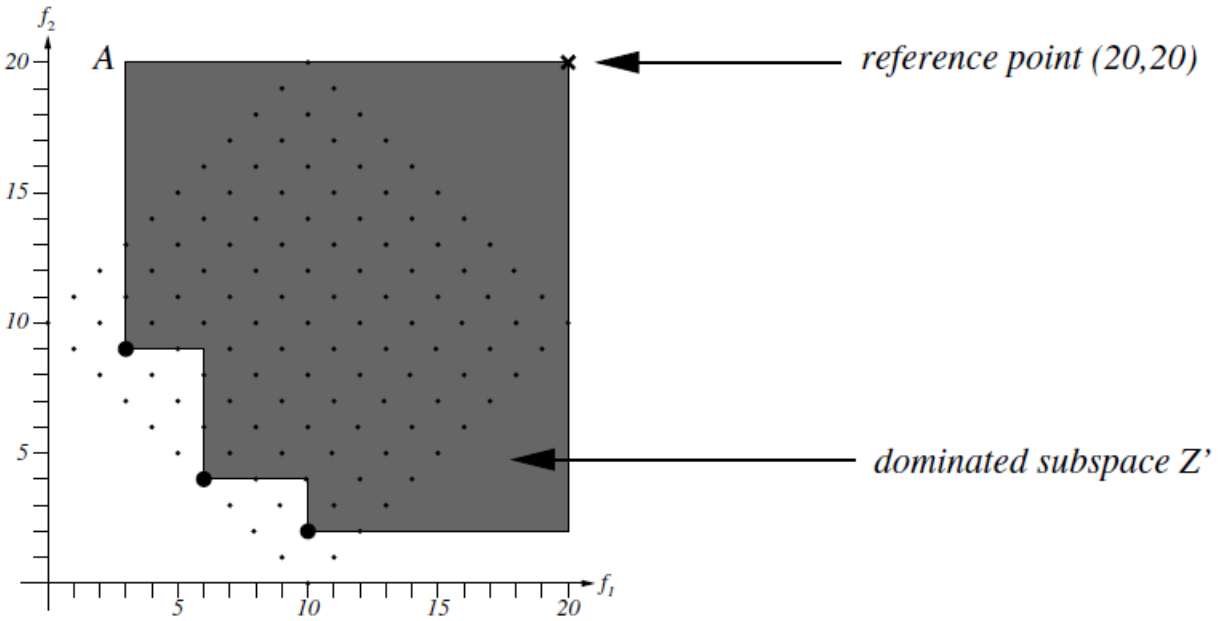
Gegeben (zu minimierende) Fitnessfunktionen $f = (f_1, \dots, f_n)$ und ein approximation set A, berechnet sich das Hypervolumen (hypervolume) als

$$I_{\mathrm{H}}(A) = \int_{z_{\mathrm{low}}}^{z_{\mathrm{up}}} \alpha_A(z) dz.$$

Hierbei sind z_{low} und z_{up} die untere und obere Schranke, innerhalb derer das Hypervolumen berechnet werden soll, und

$$\alpha_A(z) = \begin{cases} 1, \exists x \in A: f(x) \leq z \\ 0, \mathrm{sonst} \end{cases}.$$

Illustration Hypervolumen



Quelle: [ZKT08]

Hypervolumen für das approximation set A aus den vorherigen Beispielen. Auf eine untere Schranke kann (bei Minimierungsproblemen) verzichtet werden; eine obere Schranke muss gewählt werden, um ein endliches Hypervolumen zu erhalten (für Maximierungsprobleme umgekehrt).

Eigenschaften Hypervolumen

Eigenschaften:

- Streng monoton
- Nicht invariant gegenüber Skalierung der Fitnessfunktionen
- Aufwändig zu berechnen (Algorithmen zur Berechnung haben exponentielle Laufzeit in der Anzahl der Optimierungsziele)
- Abhängig von einem reference point (obere bzw. untere Schranke, bis zu der das Volumen berechnet wird, abhängig davon, ob Minimierungs- oder Maximierungsproblem)

Literatur

- [ZKT08] Eckart Zitzler, Joshua D. Knowles, Lothar Thiele: Quality Assessment of Pareto Set Approximations, in: Jürgen Branke, Kalyanmoy Deb et al.: Multiobjective Optimization. Interactive and Evolutionary Approaches, Springer 2008, 373–404