

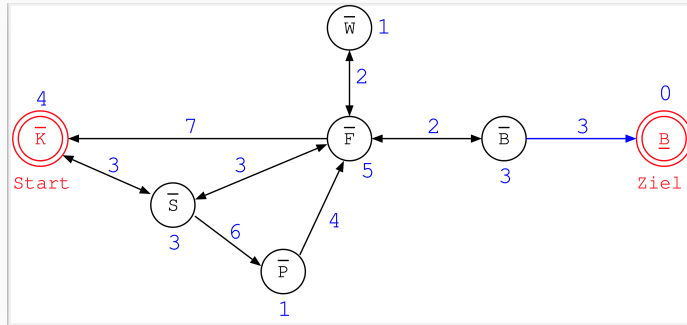
# Suche mit $A^*$

---

Carsten Gips (FH Bielefeld)

Unless otherwise noted, this work is licensed under CC BY-SA 4.0.

# Hole das Buch



**Informierte Suche: Nutzung der Kostenfunktion:**

**Gesamtkosten:**  $f(n) = g(n) + h(n)$

- Kombination aus Branch-and-Bound und Best-First-Suche
- Kostenfunktion:  $f(n) = g(n) + h(n)$
- Datenstruktur: **sortierte Queue** (Prioritätsqueue)
- Voraussetzung:
  1. Alle Aktionen haben positive Kosten ( $g(n) \geq \epsilon$ )
  2. Heuristik  $h(n)$  muss *zulässig/konsistent* sein

# A\*-Suche – Anforderungen an Heuristik (Tree-Search)

**Tree-Search-Variante:** Die Heuristik muss **zulässig** sein:

- Seien  $h^*(n)$  die tatsächlichen optimalen Restkosten von einem Knoten  $n$  zum nächsten Ziel.
- Dann muss für jeden beliebigen Knoten  $n$  gelten:

$$h(n) \leq h^*(n)$$

- Außerdem muss gelten:
  - $h(n) \geq 0$  für jeden Knoten  $n$
  - $h(n) = 0$  für jeden Zielknoten  $n$

=> Beispiel: Luftlinie als Abschätzung

# A\* ist optimal

A\* (Tree-Search-Variante) mit zulässiger Heuristik ist optimal.

Beweis siehe Übungsblatt “Blatt 01” :-)

## Einfache Verbesserungen A\* (Tree-Search)

- Dynamische Programmierung: Behalte von mehreren Pfaden zum gleichen Knoten nur den günstigsten in der Queue
- Pfade, deren Endknoten bereits früher im Pfad vorkommt (Schleifen), werden in Schritt 2 nicht in die Queue aufgenommen
- Übergang zur Graph-Search-Variante und Markierung von Knoten  
=> Achtung: Dann schärfere Anforderungen an Heuristik (Konsistenz)

# A\*-Suche – Anforderungen an Heuristik (Graph-Search)

**Graph-Search-Variante:** Die Heuristik muss **konsistent** sein:

Für jeden Knoten  $n$  und jeden durch eine Aktion  $a$  erreichten Nachfolger  $m$  gilt:

$$h(n) \leq c(n, a, m) + h(m)$$

mit  $c(n, a, m)$  Schrittkosten für den Weg von  $n$  nach  $m$  mit Aktion  $a$ .

Außerdem muss gelten:

- $h(n) \geq 0$  für jeden Knoten  $n$
- $h(n) = 0$  für jeden Zielknoten  $n$

=> Eine konsistente Heuristik ist gleichzeitig zulässig.

# Eigenschaften Branch-and-Bound, Best-First, A\*

	Branch-and-Bound	Best-First	A*
Kosten	$f(n) = g(n)$	$f(n) = h(n)$	$f(n) = g(n) + h(n)$
Vollständigkeit	ja <sup>1</sup>	nein <sup>2</sup>	ja
Optimalität	ja	nein	ja
Aufwand	exponentiell	exponentiell	exponentiell
Bemerkung	Probiert erst alle "kleinen" Pfade	Suchverlauf stark abh. v. Heuristik	Heuristik: zulässig bzw. konsistent

<sup>1</sup>BnB vollständig: Kosten größer Epsilon (positiv)

<sup>2</sup>gilt für Tree-Search-Variante; vollständig bei Graph-Search und endlichen Problemräumen



- Informierte Suchverfahren
  - Nutzen reale Pfadkosten und/oder Schätzungen der Restkosten
  - A\*: komplette Kostenfunktion  $f(n) = g(n) + h(n)$ 
    - ⇒ besondere Anforderungen an die Heuristik!

# LICENSE



Unless otherwise noted, this work is licensed under CC BY-SA 4.0.