

Grundlagen der Künstlichen Intelligenz

5. Suchalgorithmen: Heuristiken

Malte Helmert

Universität Basel

18. März 2013

Einleitung

Suchprobleme: Überblick

Kapitelüberblick:

- Formalisierung von Suchproblemen
 - ~~> [Kapitel 3](#)
- blinde Suchverfahren
 - ~~> [Kapitel 4](#)
- informierte Suchverfahren
 - ~~> [ab diesem Kapitel](#)

Informierte Suchverfahren

- bisherige Suchverfahren: **blind**, da sie keine Problemeigenschaften ausser der Problemdefinition (Zustandsraum) verwenden
- **Problem:** Skalierbarkeit \rightsquigarrow schon bei intuitiv **einfachen** Problemen Zeit- und Speicherlimits erreicht
- **Idee:** versuche (problemspezifische) Kriterien zu finden, welche Zustände **gut** und welche **schlecht** sind
 \rightsquigarrow **bevorzuge gute Zustände**

\rightsquigarrow **informierte Suchverfahren**

Heuristiken

Heuristiken

Definition (Heuristik)

Sei \mathcal{S} ein Zustandsraum mit Zustandsmenge S .

Eine **Heuristikfunktion** oder **Heuristik** für \mathcal{S} ist eine Funktion

$$h : S \rightarrow \mathbb{R}_0^+ \cup \{\infty\},$$

die jedem Zustand einen Zahlenwert (oder ∞) zuordnet.

Heuristiken: Intuition

Idee: $h(s)$ schätzt Abstand von s zum Ziel

- Heuristiken können (zunächst) **beliebige Funktionen** sein
- **Intuition:** je näher h am tatsächlichen Zielabstand,
desto effizienter die Suche

Heuristiken werden gelegentlich über **Suchknoten** (statt Zuständen)
definiert, aber diese Allgemeinheit ist selten nützlich. ([Warum?](#))

Notation: für Suchknoten n schreiben wir zur Vereinfachung $h(n)$
für $h(n.state)$

Wieso „Heuristik“?

Was heisst „Heuristik“?

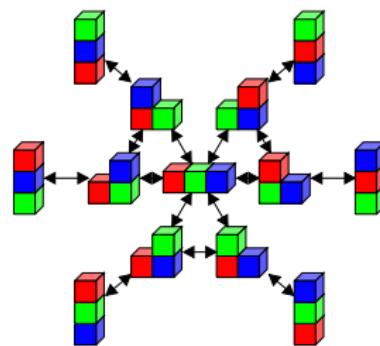
- Heuristik: aus altgriechisch ἔυρισκω (= ich finde)
siehe auch: ἔυρηκα!
- Begriff in der modernen Wissenschaft popularisiert durch George Pólya: How to Solve It (1945)
- oft verwendet für: Daumenregel/unpräzise Methode
- in der Suche präziser Begriff für Zielabstandsschätzung

Beispiele

Beispiel: Blocks world

mögliche Heuristik:

+1 für jeden Block x , der auf y liegt
und im Ziel auf $z \neq y$ liegen muss
(inkl. dem Fall, dass y oder z der Tisch ist)

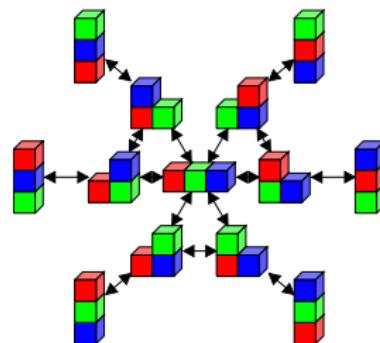


Beispiel: Blocks world

mögliche Heuristik:

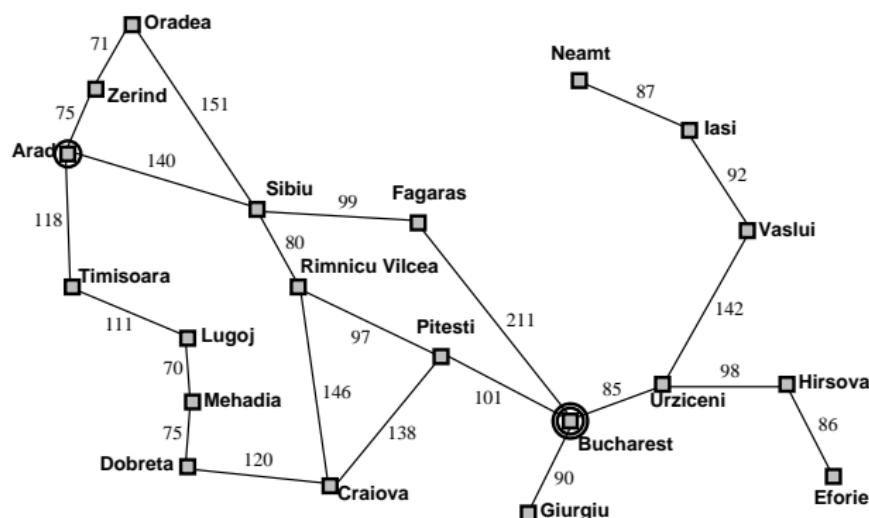
+1 für jeden Block x , der auf y liegt
und im Ziel auf $z \neq y$ liegen muss
(inkl. dem Fall, dass y oder z der Tisch ist)

Wie genau ist diese Heuristik?



Beispiel: Routenplanung in Rumänien

mögliche Heuristik: Luftliniendistanz nach Bukarest



| | |
|----------------|-----|
| Arad | 366 |
| Bucharest | 0 |
| Craiova | 160 |
| Drobeta | 242 |
| Eforie | 161 |
| Fagaras | 176 |
| Giurgiu | 77 |
| Hirsova | 151 |
| Iasi | 226 |
| Lugoj | 244 |
| Mehadia | 241 |
| Neamt | 234 |
| Oradea | 380 |
| Pitesti | 100 |
| Rimnicu Vilcea | 193 |
| Sibiu | 253 |
| Timisoara | 329 |
| Urziceni | 80 |
| Vaslui | 199 |
| Zerind | 374 |

Beispiel 3: Missionare und Kannibalen

Aufgabe: Missionare und Kannibalen

- sechs Personen müssen einen Fluss überqueren
- sie besitzen ein Boot, mit dem ein oder zwei Personen über den Fluss rudern können (mehr passen nicht hinein)
- drei der Personen sind Missionare, drei sind Kannibalen
- Missionare dürfen nicht mit einer Mehrheit an Kannibalen allein gelassen werden

mögliche Heuristik: Anzahl Personen am falschen Ufer

Beispiel 3: Missionare und Kannibalen

Aufgabe: Missionare und Kannibalen

- sechs Personen müssen einen Fluss überqueren
- sie besitzen ein Boot, mit dem ein oder zwei Personen über den Fluss rudern können (mehr passen nicht hinein)
- drei der Personen sind Missionare, drei sind Kannibalen
- Missionare dürfen nicht mit einer Mehrheit an Kannibalen allein gelassen werden

mögliche Heuristik: Anzahl Personen am falschen Ufer

- ~~> bei unserer Formalisierung als Tripel $\langle m, c, b \rangle$:
- $$h(\langle m, c, b \rangle) = m + c$$

Eigenschaften von Heuristiken

Perfekte Heuristik

Definition (perfekte Heuristik)

Sei \mathcal{S} ein Zustandsraum mit Zustandsmenge S .

Die **perfekte Heuristik** für \mathcal{S} , geschrieben h^* , ordnet jedem Zustand $s \in S$ die Kosten einer **optimalen Lösung** für s zu.

Anmerkung: $h^*(s) = \infty$, falls s Sackgasse

Eigenschaften von Heuristiken

Definition (sicher, zielerkennend, zulässig, konsistent)

Sei \mathcal{S} ein Zustandsraum mit Zustandsmenge S .

Eine Heuristik h für \mathcal{S} heisst

- **sicher**, falls $h^*(s) = \infty$ für alle $s \in S$ mit $h(s) = \infty$
- **zielerkennend**, falls $h(s) = 0$ für alle Zielzustände s
- **zulässig**, falls $h(s) \leq h^*(s)$ für alle Zustände $s \in S$
- **konsistent**, falls $h(s) \leq cost(a) + h(s')$ für alle Transitionen $s \xrightarrow{a} s'$

englisch: safe, goal-aware, admissible, consistent

Eigenschaften von Heuristiken: Beispiele

Welche der Beispielheuristiken haben welche dieser Eigenschaften?

Routenplanung:

- sicher
- zielerkennend
- zulässig
- konsistent

Warum?

andere Beispiele:

~~~ Hausaufgaben

# Eigenschaften von Heuristiken: Zusammenhänge

Satz (zulässig  $\Rightarrow$  sicher + zielerkennend)

*Sei  $h$  eine zulässige Heuristik.*

*Dann ist  $h$  auch sicher und zielerkennend.*

Warum?

Satz (zielerkennend + konsistent  $\Rightarrow$  zulässig)

*Sei  $h$  eine zielerkennende und konsistente Heuristik.*

*Dann ist  $h$  auch zulässig.*

Warum?

Wie beweist man am einfachsten, dass eine Heuristik alle vier Eigenschaften hat?

# Wie kommen wir zu Heuristiken?

# Wie kommt die Heuristik in den Computer?

Wie kommt die Heuristik in den Computer?

Erinnerung: drei Repräsentationen für Zustandsräume

- ① als expliziter Graph
- ② als deklarative Beschreibung
- ③ als Black Box

# Heuristiken für verschiedene Arten von Zustandsräumen

Wie kommt die Heuristik in den Computer?

## ① Zustandsraum als expliziter Graph:

Knoten (Zustände) und Kanten (Transitionen) explizit repräsentiert,  
z. B. über **Adjazenzlisten** oder als **Adjazenzmatrix**

- ~~> Heuristik ebenfalls als **explizite Tabelle** repräsentiert  
(evtl. partiell oder komprimiert)

# Heuristiken für verschiedene Arten von Zustandsräumen

Wie kommt die Heuristik in den Computer?

## ② Zustandsraum als deklarative Beschreibung:

Kompakte Beschreibung des Zustandsraums ist Input des Solvers.

- ~~> Heuristik wird **automatisch aus der Beschreibung** des Zustandsraums erzeugt
- ~~> später

# Heuristiken für verschiedene Arten von Zustandsräumen

Wie kommt die Heuristik in den Computer?

- ③ als Black Box: abstraktes Interface für Zustandsräume

Abstraktes Interface für Zustandsräume mit Heuristik

Zustandsraum als Black Box:

- $\text{init}()$
- $\text{is-goal}(s)$
- $\text{succ}(s)$
- $\text{cost}(a)$

~~~ zusätzliches Element:

- $h(s)$: Heuristikwert für Zustand s
Rückgabewert: nicht-negative Zahl oder ∞

Zusammenfassung

Zusammenfassung

- **Heuristiken** schätzen Abstand eines Zustands zum Ziel
- Heuristiken fokussieren die Suche

wichtige Eigenschaften von Heuristiken:

- sicher
- zielerkennend
- zulässig
- konsistent