Der Wert des Information-Sammelns

Zusätzliche Information hat Wert in dem Maß wie sie das Handeln ändert, um den erwarteten Gesamtnutzen zu steigern

Erwarteter Nutzen bei gegebener Evidenz E:

$$EU(A|E) = \max_a \sum_i P(\text{Result}_i(a) \mid \text{Do}(a), E) \times U(\text{Result}_i(a))$$

Erwarteter Nutzen für *neue* Evidenz in Variable E_j:

$$EU(A_{Ej}|E, E_j) = \max_a \sum_i P(\text{Result}_i(a) \mid \text{Do}(a), E, E_j) \times U(\text{Result}_i(a))$$

(Erwarteter) „Wert der *perfekten* Information“ (**VPI**) bzgl. E_j :
Berechne erwarteten Nutzengewinn über alle möglichen Ausprägungen e_{jk} von E_j:

$$VPI_E(E_j) = \left[\sum_k P(E_j = e_{jk} \mid E) \times EU(A_{ejk}|E, E_j{=}e_{jk})\right] - EU(A|E)$$

Eigenschaften von VPI

Nichtnegativ $\forall j, E \;\; VPI_E(E_j) \geq 0$

Achtung: VPI ist der *erwartete* Wert!

Nichtadditiv $VPI_E(E_j, E_k) \neq VPI_E(E_j) + VPI_E(E_k)$

Beispiel für Ungleichheit: $E_j = E_k$

Kommutativ

$$VPI_E(E_j, E_k) = VPI_E(E_j) + VPI_{E,E_j}(E_k) = VPI_E(E_k) + VPI_{E,E_k}(E_j)$$

Informationsbeschaffung als *sequenzielles* Problem

Entscheiden mit Informationsammeln

function INFORMATION-GATHERING-AGENT(*percept*) **returns** an *action*
  static: D, a decision network

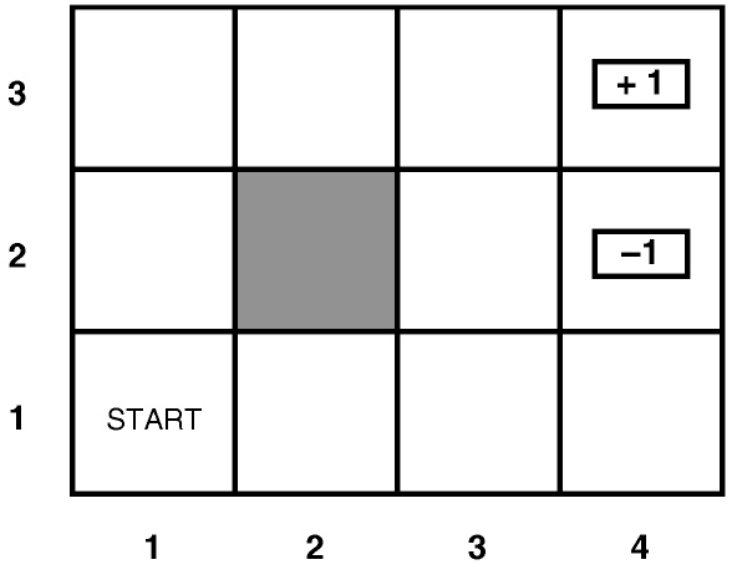
  integrate *percept* into D
  $j \leftarrow$ the value that maximizes $VPI(E_j) - Cost(E_j)$
  if $VPI(E_j) > Cost(E_j)$
    then return REQUEST(E_j)
  else return the best action from D

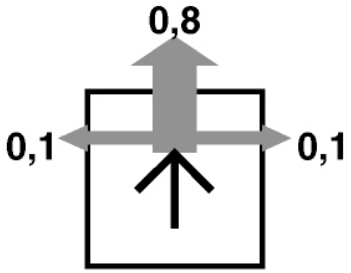
Als nächstes gehen wir von der Entscheidung für *einzelne* Aktionen zum Generieren von *Plänen* unter Unsicherheit

Sequenzielle Entscheidungsprobleme

Ab jetzt: Optimale *Sequenzen* von Aktionen unter Unsicherheit

Beispiel





Up(1,1) = [0,8,⟨1,2⟩; 0,1,⟨2,1⟩; 0,1,⟨1,1⟩]
Right(,)=[…] **Left**(,)=[…] **Down**(,)=[…]

Up, **Up**, **Right**, **Right**, **Right** erreicht ⟨4,3⟩ mit
$P = 0{,}8^5 + 0{,}1^4 \times 0{,}8 = 0{,}32768 + 0{,}00008 = 0{,}32776$

Markowsche Entscheidungsprozesse (MDPs)

Voraussetzungen:

- Vollständige Beobachtbarkeit
  (Agent ermittelt sicher, auf welcher Position er ist)
- Markow-Eigenschaft: Folgezustand hängt nur ab von Ausgangszustand und Aktion (nicht von früheren Aktionen)

MDP:

- **Startzustand** S_0 (Beispiel: $\langle 1,1 \rangle$)
- **Transitionsmodell** $T(s,a,s')$: W'keit, von Zustand s mit a in s' zu kommen (syntaktische Variante der entspr. Lotterie)
  (Beispiel: $T(\langle 1,1 \rangle, \mathbf{Up}, \langle 1,2 \rangle)=0,8$)
- ***Reward*-Funktion** $R(s)$: Belohnung (positiv oder negativ), einen Zustand zu erreichen (Beispiel: –0,04 außer für $\langle 4,2 \rangle$, $\langle 4,3 \rangle$)

Der Nutzen des Agierens

Voraussetzungen:

- Nutzen ergibt sich aus der Summe von *Rewards* der besuchten Zustände
- *Rewards* in naher Zukunft sind möglicherweise anders zu gewichten als in ferner Zukunft: Faktor $0 \leq \gamma \leq 1$ (Abschlag, *discount factor*)
- Die Länge von Aktionssequenzen ist a priori nicht beschränkt

$$U([s_0, s_1, s_2, \ldots]) = \sum_{t=0}^{\infty} \gamma^t R(s_t)$$

Für $\gamma < 1$ und $R_{\max}$ ist der Nutzen jeder Aktionssequenz endlich:

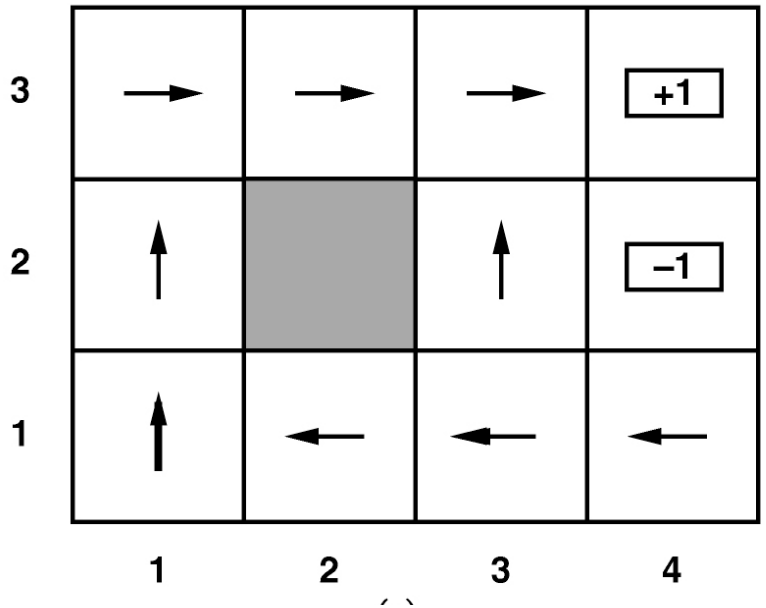
$$U([s_0, s_1, s_2, \ldots]) = \sum_{t=0}^{\infty} \gamma^t R(s_t) \leq \sum_{t=0}^{\infty} \gamma^t R_{\max} = \frac{R_{\max}}{1-\gamma}$$

MDP-Pläne (Taktiken, Politiken, *policies*)

Nutzen von Aktionssequenzen bewertet *beobachtetes* Verhalten, hilft aber nicht entscheiden, was der Agent in Zustand s_i tun soll!

Ein **MDP-Plan** ist eine Abbildung $\pi : S \rightarrow A$

Beispiel



Ein **optimaler MDP-Plan** ist ein MDP-Plan mit maximalem erwartetem Nutzen:

$$\pi^* = \underset{\pi}{\operatorname{argmax}}\, E\left[\sum_{t=0}^{\infty} \gamma^t R(s_t) \middle| \pi\right]$$

Zwei Basisalgorithmen dafür: *Value Iteration, Policy Iteration*

„Rückrechnen“ von Nutzen auf Zustände

Grundidee der *Value Iteration*:

- Definiere **Bewertung** von Zuständen: Was erwarte ich, dass ich langfristig als Reward bekomme, wenn ich in s bin?
- Bewertung heißt **Nutzen** $U^{\pi}(s)$
  (**Vorsicht Falle**: Die Reward-Funktion des MDP bleibt unverändert!)
- MDP-Plan macht dann Gradientenaufstieg über erw. Nutzen

$$U^{\pi}(s) := E\left[\sum_{t=0}^{\infty} \gamma^t R(s_t) \middle| \pi, s_0 = s\right]$$

Der „objektive Wert“ eines Zustands ist dann der Nutzen eines optimalen MDP-Plans: $U(s) = U^{\pi^*}(s)$, oder anders …

Die Bellmann-Gleichung

Der Nutzen eines Zustands ist sein *Reward* plus der (erwartete, diskontierte) Nutzen des Nachfolgezustands unter der optimalen Folgeaktion

$$U(s) = R(s) + \gamma \max_a \sum_{s'} T(s,a,s')U(s')$$

… definiert ein Gleichungssystem über ein MDP, das $U^{\pi^*}(s)$ lokal charakterisiert

Idee der *Value Iteration*: Approximiere $U(s)$ durch lokale Propagierung von Nutzenwerten bzw. *Rewards* an Nachbarzustände gemäß Transitionsmodell T des MDP, bis Werte „hinreichend stabil“

Value Iteration

function VALUE-ITERATION(mdp, ϵ) **returns** a utility function
  inputs: mdp, an MDP with states S, transition model T, reward function R, discount γ
    ϵ, the maximum error allowed in the utility of any state
  local variables: U, U', vectors of utilities for states in S, initially zero
    δ, the maximum change in the utility of any state in an iteration

  repeat
    $U \leftarrow U'; \delta \leftarrow 0$
    for each state s **in** S **do**
      $U'[s] \leftarrow R[s] + \gamma \max_a \sum_{s'} T(s, a, s') U[s']$
      if $|U'[s] - U[s]| > \delta$ **then** $\delta \leftarrow |U'[s] - U[s]|$
  until $\delta < \epsilon(1 - \gamma)/\gamma$
  return U

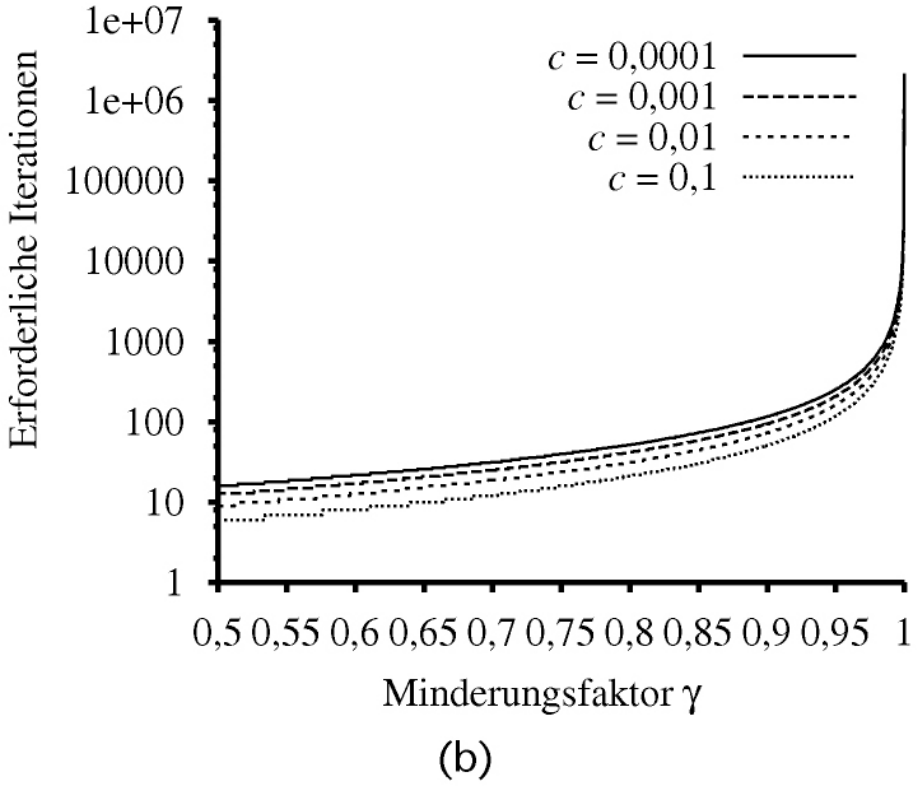
Konvergenz der *Value Iteration*

Satz

Der *Value Iteration*-Algorithmus konvergiert für alle Zustände s des MDP auf den Nutzen $U^{\pi^*}(s)$, welcher dem optimalen MDP-Plan π^* entspricht

Beweisskizze: s. Russell/Norvig 17.2

Zahl der Iterationen bis zur Konvergenz in Abhängigkeit vom Abschlagsfaktor γ und vom erlaubten Fehler $\varepsilon = c \cdot R_{\max}$

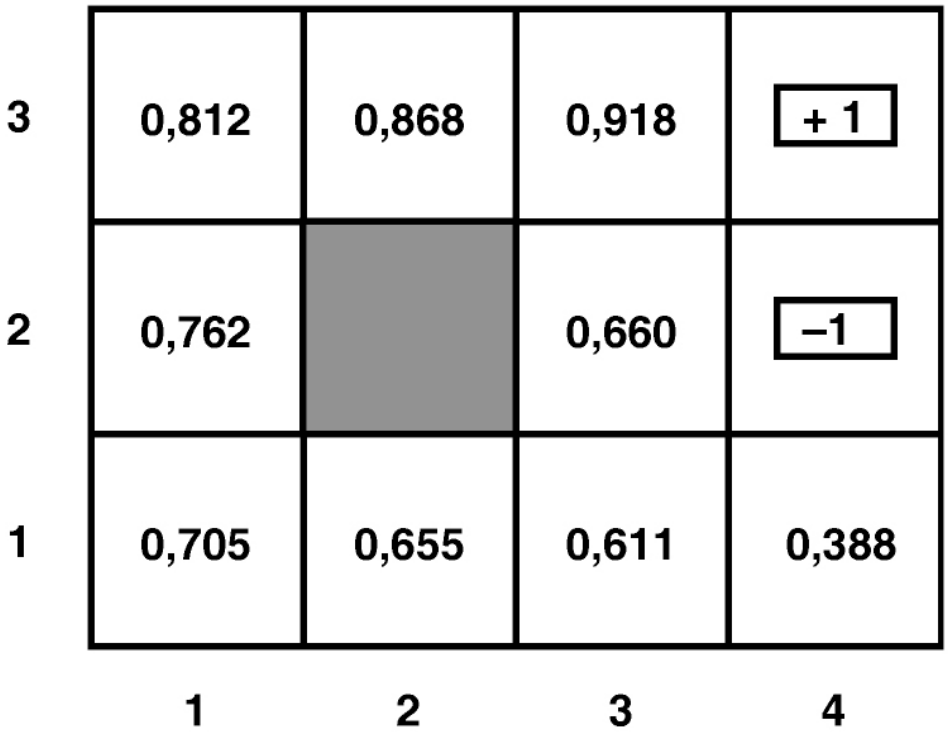


(b)

Ergebnis für Beispiel-MDP lt. Russell/Norvig

angeblich:

- $\gamma=1$,
- $R(s)=-0{,}04$ für nichtterminale s



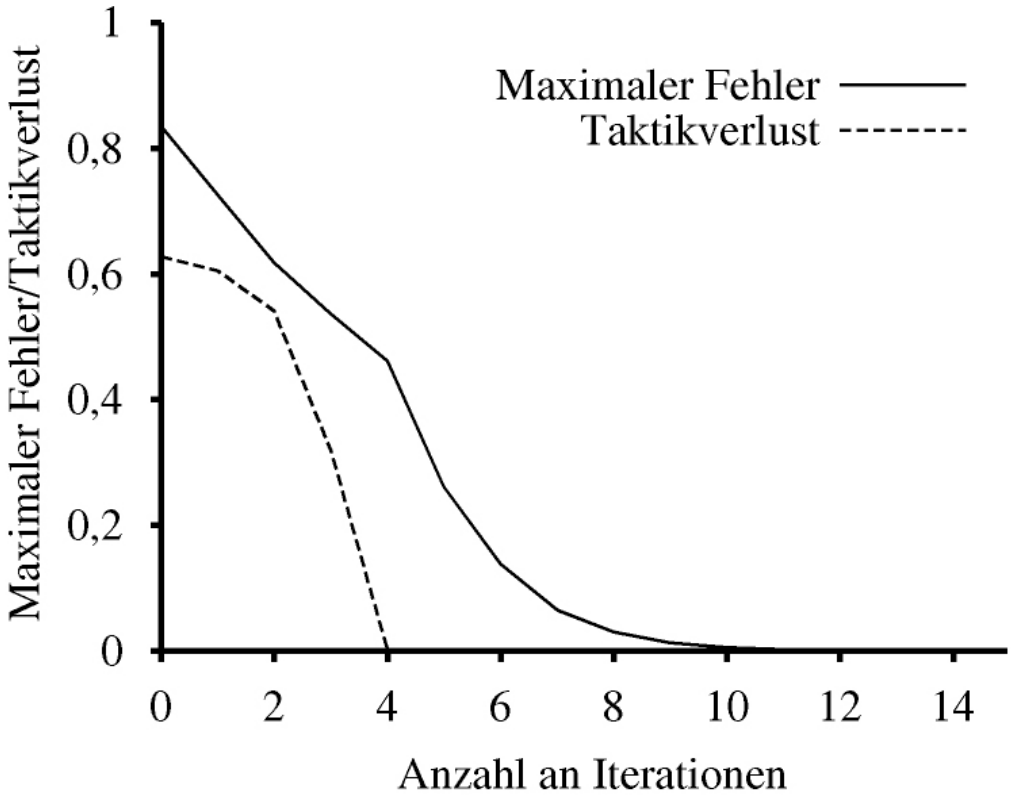
Probe:

$$U(\langle 3,3\rangle) = R(\langle 3,3\rangle) + 1\text{x}\sum T(\langle 3,3\rangle, \textbf{Right}, s')U(s')$$
$$= -0{,}04 + 0{,}8\text{x}1 + 0{,}1\text{x}0{,}660 + 0{,}1\text{x}0{,}918$$
$$= -0{,}04 + 0{,}8 + 0{,}066 + 0{,}0918 = 0{,}9178$$

Achtung: Probe klappt nicht für andere Zustände!! (z.B. $\langle 3,1\rangle$)

Ist das wirklich schlimm?

Gesucht ist nicht der „wahre“, objektive Nutzen eines Zustands, sondern der optimale MDP-Plan!



Der optimale MDP-Plan ergibt sich zumeist schon bei recht grob approximierten Utilities!

Optimale MDP-Pläne ohne präzise Utilities

Grundidee der *Policy Iteration*: Starte mit beliebigem (zufällig gewähltem) MDP-Plan, iteriere die folgenden beiden Schritte:

- **Bewertung**: Berechne die Utility U_i jedes Zustands unter dem aktuellen MDP-Plan π_i
- **Verbesserung**: Berechne, wenn möglich, basierend auf den aktuellen Utilities einen besseren MDP-Plan π_{i+1}

Effizienter als *Value Iteration*, weil für Bewertung aktuelle Utility der Zustände nicht über *alle* möglichen Aktionen maximiert werden muss! Vereinfachung der Bellmann-Gleichung (n Gleich. mit n Unbekannten, lösbar in $O(n^3)$):

$$U_i(s) = R(s) + \gamma \sum_{s'} T(s,\pi_i(s),s')U_i(s')$$

Für große n: Berechne für aktuelle Schätzung der $U_i(s)$ in k Durchläufen (Komplexität: kn):

$$U_{i+1}(s) = R(s) + \gamma \sum_{s'} T(s,\pi_i(s),s')U_i(s')$$

Policy Iteration

function POLICY-ITERATION(mdp) **returns** a policy
  inputs: mdp, an MDP with states S, transition model T
  local variables: U, U', vectors of utilities for states in S, initially zero
    π, a policy vector indexed by state, initially random

  repeat
    $U \leftarrow$ POLICY-EVALUATION(π, U, mdp)
    $unchanged? \leftarrow$ **true**
    for each state s **in** S **do**
      if $\max_a \sum_{s'} T(s,a,s')\ U[s'] > \sum_{s'} T(s,\pi[s],s')\ U[s']$ **then**
        $\pi[s] \leftarrow \text{argmax}_a \sum_{s'} T(s,a,s')\ U[s']$
        $unchanged? \leftarrow$ **false**
  until $unchanged?$
  return π

POLICY-EVALUATION ist die Bewertung auf der vorigen Folie

Partielle Beobachtbarkeit: POMDPs

Voraussetzungen im Unterschied zu MDPs:

- Keine vollständige Beobachtbarkeit (Agent weiß evtl. nicht, auf welcher Position er ist), außer möglicherweise bei Ziel
- **Sensormodell** $O(s,o)$: W'verteilung; gibt an, mit welcher W'keit in s die Beobachtung (*observation*) o gemacht wird; ändert nicht den Zustand!

  Bspl.: Steht Agent in Blickrichtung vor einer Wand, nimmt er das mit p=0,7 wahr; mit p=0,1 sieht er eine Wand, wo keine ist.

POMDP (MDP unter partieller Beobachtbarkeit):

- Zustandsmenge (Start S_0), $T(s,a,s')$, $R(s)$ wie in MDP
- **Sensormodell** $O(s,o)$
- **Überzeugungszustand** (*belief state,* Subjektive Zustandsinformation): W'verteilung $b(s)$ über der Zustandsmenge (*abgeleitete* Komponente!)

POMDP Beispiel

Überzeugungszustand bei Gleichverteilung über die Nicht-Zielzustände:

$b = \langle 1/9, 1/9, \ldots, 1/9, 0, 0 \rangle$

$b(\langle 1,1 \rangle) = 1/9, \quad b(\langle 4,3 \rangle) = 0$

0,111	0,111	0,111	0,000
0,111		0,111	0,000
0,111	0,111	0,111	0,111

Werte „Überzeugungs-Masse“ (Aufenthaltsw‘keit), **nicht Nutzen**!

Aktualisierung des Überzeugungszustands

- Neuer Ü-Zustand b' wird errechnet, nachdem Aktion a ausgeführt und eine Sensormessung o im resultierenden Weltzustand gemacht wurde (normalisiere b' mit Faktor α):

$$b'(s') = \alpha O(s',o) \sum_s T(s,a,s') b(s)$$

Erweitert auf die Zustandsmenge: $b' := \text{FORWARD}(b,a,o)$

Wiederholtes Aktualisieren des Ü-Zustands nach Beobachtung ohne Aktion führt zu Verletzung der Unabhängigkeitsannahme! („solange beobachten, bis die Überzeugung konvergiert“)

Mangels Beobachtbarkeit des „objektiven“ Weltzustands basiere die Aktion auf den Ü-Zustand!

Ü-MDPs (*belief* MDPs)

POMDP-Pläne können bestimmt werden als optimale MDP-Pläne im zugehörigen Ü-MDP (*belief* MDP)!

Startzustand des Ü-MDP: Initialer Ü-Zustand

***Reward*-Funktion des Ü-MDP** $\rho(b) = \sum_s b(s)R(s)$

Transitionsmodell des Ü-MDP

$$= \sum_{s'} P(o \mid a,s',b)P(s' \mid a,b) = \sum_{s'} O(s',o) \sum_s T(s,a,s')b(s)$$

$$\tau(b,a,b') = P(b' \mid a,b) = \sum_o P(b' \mid o,a,b) P(o \mid a,b)$$

1, wenn b' = FORWARD(b,a,o), 0 sonst

Optimale Ü-MDP-Pläne

… kann man *im Prinzip* mit *Value/Policy Iteration* finden,

… doch sind Ü-MDPs in der Regel hochdimensional und haben immer einen kontinuierlichen Zustandsraum!
Bspl.: Ü-MDP für 11er-Giter 11-dimensional

- Russell/Norvig stellen kurz *dynamic decision networks* vor (DDNs, eine Variante von Bayes-Netzen)
- Einige Arbeiten (einschließlich eigenen) existieren dazu, die Ü-Zustände als Mengen propositionaler Fakten zu repräsentieren (Ü-Zustände als W'keit möglicher Modelle)

Es gibt derzeit kein wirklich praktikables Verfahren, POMDPs interessanter Größe zu behandeln!

Handlungsplanung

Mehr dazu

- Russell/Norvig
- M. Ghallab, D. Nau & P. Traverso:
  Automated Planning – Theory and Practice
  Morgan Kaufmann, 2004
- http://www.planet-noe.org/ > Service > Repositories
  Sammlung von Informationen und Links
  (einschließlich herunterladbare Planungssysteme u.v.a.)
- Seminar im SS 2005