Sekvenční rozhodování za neurčitosti

Jiří Kléma

Katedra kybernetiky,
FEL, ČVUT v Praze

IDA
Intelligent Data Analysis
RESEARCH GROUP

http://cw.felk.cvut.cz/doku.php/courses/a4b33zui/start

Plán přednášky

- Minule: rozhodování za neurčitosti
  - preference mezi stavy resp. funkce užitku,
  - stochastické výsledky akcí – loterie, střední užitek,
- jak optimálně volit celé posloupnosti akcí?
  - opakovaná rozhodnutí s nepřesnou či nedostatečnou informací,
  - prémie/odměna je často odložená,
  - svět nemusí být plně pozorovatelný,
- markovský rozhodovací proces
  - zavádí markovský předpoklad – omezení paměti procesu,
  - pracuje i s předpoklady stacionarity a pozorovatelnosti,
  - svět je známý a popsatelný přechodovou funkcí a funkcí odměny,
- zobecnění
  - pouze částečná pozorovatelnost světa – POMDP,
  - model prostředí ani funkce odměny není k dispozici – posilované učení.

Markovův proces

- náhodný proces s pravděpodobnostmi navštěvy dalších stavů danými jen stavy nedávnými,
- vzdálená minulost je irelevantní, známe-li tu blízkou,
- **Markovův** řetězec
  - diskrétní náhodný proces s markovovskou vlastností,
  - **řád** m řetězce určuje kolik minulých stavů musíme zohlednit

    $$Pr(X_n = x_n | X_{n-1} = x_{n-1}, \ldots, X_1 = x_1) = Pr(X_n = x_n | X_{n-1} = x_{n-1}, \ldots, X_{n-m} = x_{n-m})$$

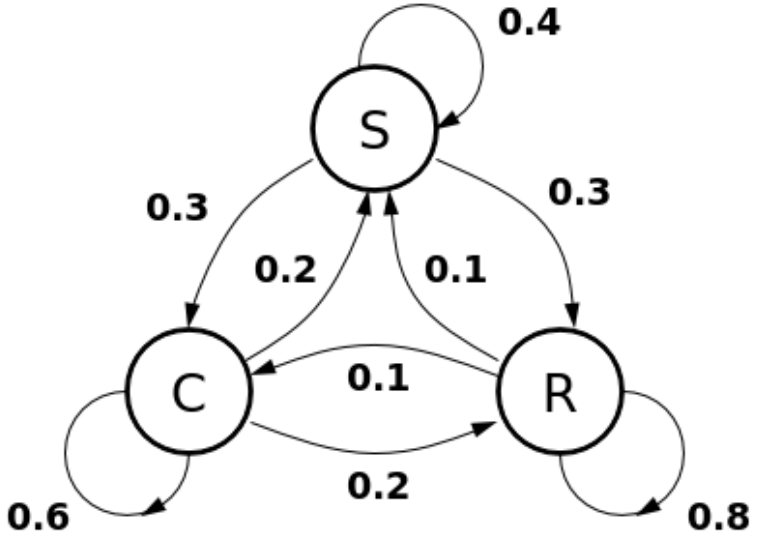
  - nejobvyklejší jsou modely řádu 1,
  - obvykle i předpoklad stacionarity (neměnnosti v čase)
    $Pr(X_{n+1} = x | X_n = y) = Pr(X_n = x | X_{n-1} = y)$
- příklady Markovových řetězců
  - hody mincí – HTHHHT ...
    - degenerovaný markovův řetězec řádu 0,
  - počasí pozorované každý den v poledne – SSSCRRCSRR ...
    - kategorizované (S)unny, (C)loudy, (R)ain, řád neznámý.

Markovův řetězec – příklad s počasím

- Jak sestavit model na základě pozorované sekvence?
  - mějme sekvenci pozorování délky 41, předpokládejme řád 1
    S S C C R R R R R R R R S S S C R R R R R R R C C C S C S S R R S R R R C S C C C
  - cílem je přechodová matice, která model definuje

$$
\begin{array}{c} \\ S_t \\ C_t \\ R_t \end{array}
\begin{array}{c}
\begin{array}{ccc} S_{t+1} & C_{t+1} & R_{t+1} \end{array} \\
\begin{bmatrix} 4 & 3 & 3 \\ 2 & 6 & 2 \\ 2 & 2 & 16 \end{bmatrix}
\end{array}
$$

$$
A = \{a_{ij}\} = \begin{bmatrix} 0.4 & 0.3 & 0.3 \\ 0.2 & 0.6 & 0.2 \\ 0.1 & 0.1 & 0.8 \end{bmatrix}
$$



Markovův řetězec – příklad s počasím

- Jaké otázky můžeme zodpovídat máme-li k dispozici Markovův model řetězce?

  1. jaká je pst, že počasí po příštích 7 dní bude "S S R R S C S", víme-li, že dnes je slunečno?
  2. víme-li, že model je ve známém stavu, jaká je pst, že v něm setrvá právě po d dní?
  3. jaká je střední hodnota d v jednotlivých stavech?

Markovův řetězec – příklad s počasím

- Jaké otázky můžeme zodpovídat máme-li k dispozici Markovův model řetězce?
  1. jaká je pst, že počasí po příštích 7 dní bude "S S R R S C S", víme-li, že dnes je slunečno?
  2. víme-li, že model je ve známém stavu, jaká je pst, že v něm setrvá právě po d dní?
  3. jaká je střední hodnota d v jednotlivých stavech?
- Řešení

1. $P(O|M) = P(S,S,S,R,R,S,C,S|M) =$
$= P(S) \times P(S|S) \times P(S|S) \times P(R|S) \times P(R|R) \times P(S|R) \times P(C|S) \times P(S|C) =$
$= 1 \times 0.4 \times 0.4 \times 0.3 \times 0.8 \times 0.1 \times 0.3 \times 0.2 = 2.3 \times 10^{-4}$

2. $O = \{\underbrace{Q_i, Q_i, \ldots, Q_i}_{d}, Q_j \neq Q_i\}$, $P(O|M, q_1 = Q_i) = a_{ii}^{d-1}(1 - a_{ii}) = p_i(d)$,

3. $\bar{d}_i = \sum_{d=1}^{\infty} d p_i(d) = \sum_{d=1}^{\infty} d a_{ii}^{d-1}(1 - a_{ii}) = \frac{1}{1-a_{ii}}$,
(součet aritmeticko-geometrické řady: $\sum_{k=0}^{\infty} k r^{k-1} = \frac{1}{(1-r)^2}$),
$d_S = 1.67$, $d_C = 2.5$, $d_R = 5$.

Sekvenční rozhodování za neurčitosti

- k dosažení cíle je obvykle třeba více kroků/akcí,
- pokud platí
  - protředí je nedeterministické (akce s nejistým výsledkem),
  - cílový stav je nahrazen cílem maximalizace kumulativní **odměny**,
- posloupnost akcí nelze nalézt klasickým plánováním
  - racionální agent by měl **přezkoumávat** svoje kroky v průběhu provádění akcí,
  - následné akce závisí na tom, co agent pozoruje,
  - to, co agent pozoruje, závisí na předchozích akcích,
- řešení
  - agent se zaměří na hodnocení stavů namísto přímého vytváření posloupností akcí,
  - v každém stavu pak zvolí akci vedoucí do následného stavu s nejvyšším ohodnocením.

Základní pojmy, definice problému

- Odměna R_t (reward)
  - prostý součet za epizodu: $R_t = r_{t+1} + r_{t+2} + r_{t+3} + \cdots + r_T$
  - diskontovaný součet pro nekonečné děje (γ je srážkový poměr, $0 \leq \gamma \leq 1$): $R_t = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \cdots = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}$
- Taktika $\pi_t(s, a)$
  - je mapováním mezi stavy a akcemi, vyjadřuje pst, že ve stavu s bude provedena akce a,
  - optimální taktika π^* maximalizuje celkovou odměnu R_t,
- Hodnota stavu $V^\pi(s)$
  - očekávaná (kumulativní) odměna pokud ze stavu s začnu aplikovat taktiku π

$$V^\pi(s) = E_\pi\{R_t \mid s_t = s\} = E_\pi\left\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \mid s_t = s\right\}$$

- Hodnota akce $Q^\pi(s, a)$
  - očekávaná (kumulativní) odměna aplikace a na s s pokračováním dle π

$$Q^\pi(s, a) = E_\pi\{R_t | s_t = s, a_t = a\} = E_\pi\left\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \mid s_t = s, a_t = a\right\}$$

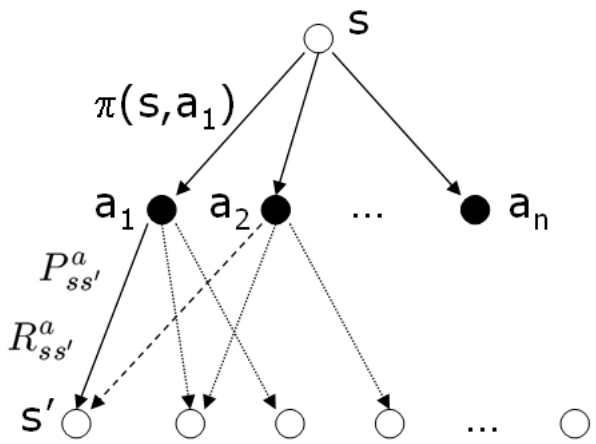
- Problém: najdi π^* (a s ní také V^*, Q^* jež jsou prostředkem).

Sekvenční rozhodování jako konečný MDP

- Konečný markovský rozhodovací proces
  - finite Markov Decision Process (MDP),
  - markovský předpoklad + množiny stavů S i akcí A jsou konečné,
  - MDP = {S,A,P,R}, lze jej zapsat jako přechodový graf,
  - P jsou přechodovými pstmi, R je funkcí odměny,

$$P^a_{ss'} = Pr\{s_{t+1} = s' \mid s_t = s, a_t = a\}$$
$$R^a_{ss'} = E\{r_{t+1} \mid s_t = s, a_t = a, s_{t+1} = s'\}$$

  - tato definice je vodítkem pro konkrétní výpočet V nebo Q.
- prostředí je popsatelné a pozorovatelné.



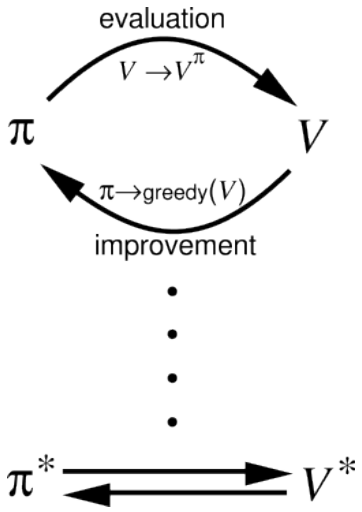
Rekurzivní definice fce V (Bellmanova rovnice)

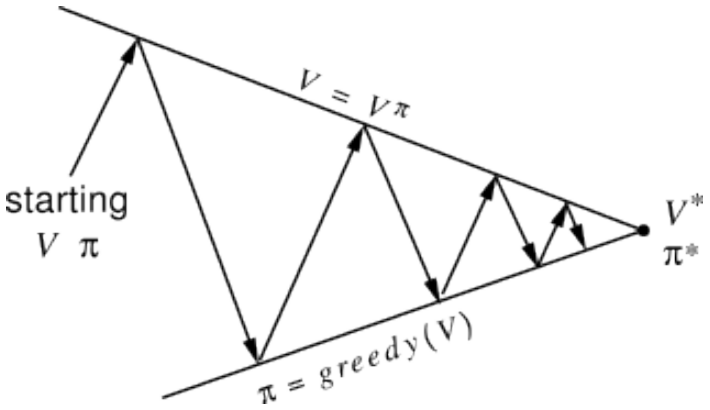
$$
\begin{aligned}
V^{\pi}(s) &= E_{\pi}\{R_t \mid s_t = s\} = \\
&= E_{\pi}\left\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \mid s_t = s\right\} = \\
&= E_{\pi}\left\{r_{t+1} + \gamma \sum_{k=0}^{\infty} \gamma^k r_{t+k+2} \mid s_t = s\right\} = \\
&= \sum_a \pi(s,a) \sum_{s'} P^a_{ss'} \left[R^a_{ss'} + \gamma E_{\pi}\left\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+2} \mid s_{t+1} = s'\right\}\right] = \\
&= \sum_a \pi(s,a) \sum_{s'} P^a_{ss'} \left[R^a_{ss'} + \gamma V^{\pi}(s')\right]
\end{aligned}
$$

- Přechod na rekurzivní definici
  - okamžitá odměna po provedení akce a očekávaná odměna pokračováním z možných následníků.
- na počátku $V^{\pi}(s)$, $V^{\pi}(s')$ i $\pi(s,a)$ neznámé
  - iterativní výpočet,
  - **bootstrapping** – postupné oživování.

Dynamické programování

- Základní přístup k řešení MDP problému (nalezení π^*)
  - dynamický = iterativní postup – pro $V(s)$ v kroku k+1 použito $V(s')$ z kroku k,
  - programování = hledání přijatelné posloupnosti akcí,
- polynomiální složitost s počtem stavů $|S|$ a akcí $|A|$
  - navzdory tomu, že počet taktik je $|A|^{|S|}$ (prohledávání SP je nutně horší),
  - přesto pro praktické úlohy často nepoužitelné
    - neznámé parametry procesu (viz RL později),
    - výpočetně nezvládnutelné (typicky příliš mnoho stavů),
      - nelze iterovat systematicky – asynchronní DP,

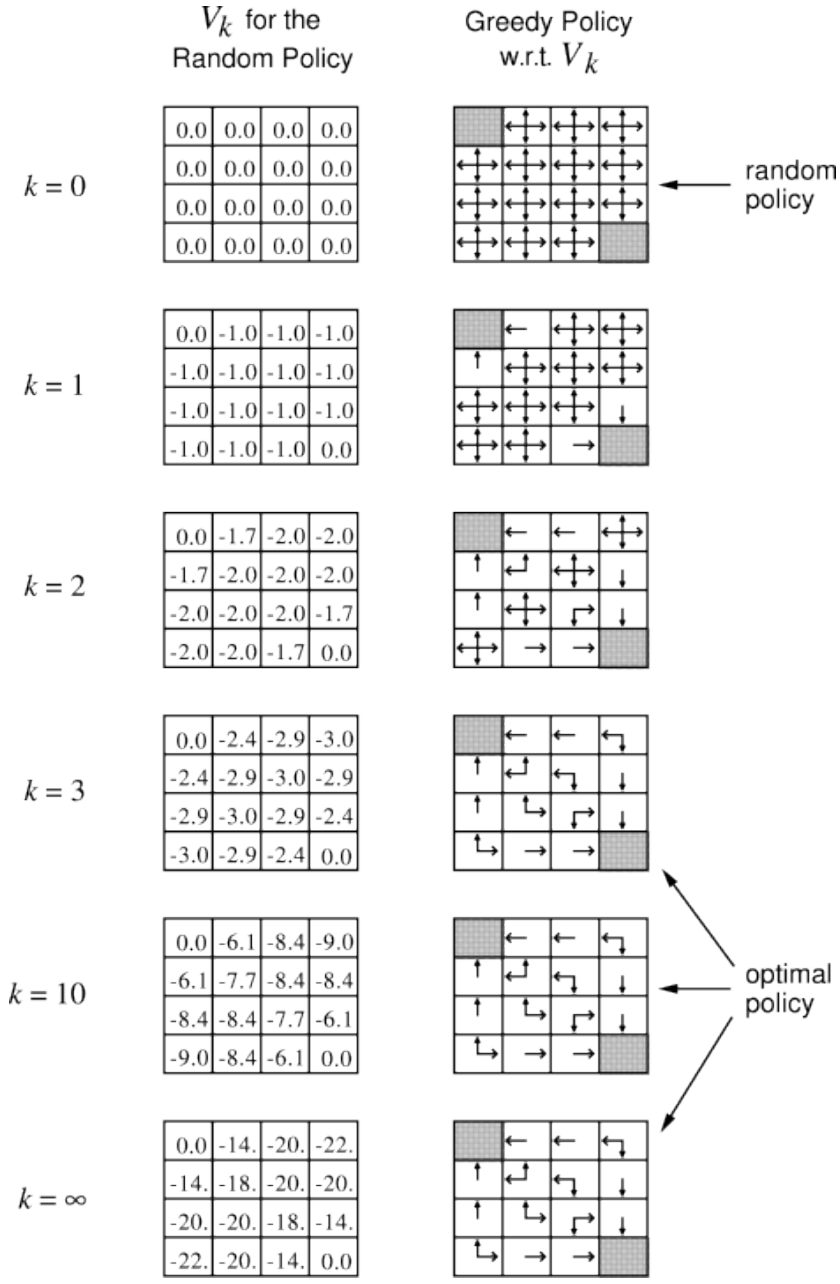


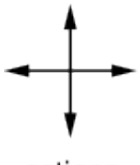


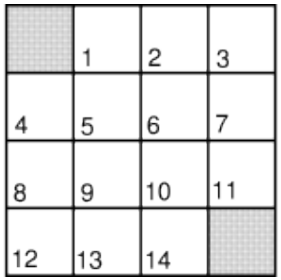
Iterace taktiky (policy iteration – PI)

- Klíčová myšlenka: $\pi_0 \xrightarrow{E} V^{\pi_0} \xrightarrow{I} \pi_1 \xrightarrow{E} V^{\pi_1} \xrightarrow{I} \ldots \xrightarrow{I} \pi^* \xrightarrow{E} V^{\pi^*}$
- využití V (popř. Q) k nalezení π^*

1. vyhodnocení taktiky π (E krok):
   - nalezení hodnot stavů $V^\pi(s)$,
   - bootstrapping, start: $V(s) = 0$ pro $\forall$ neterminální stavy (u nich je V známo),
   - iteruj dokud se hodnoty V neustálí ($\max_S |V_{k+1}(s) - V_k(s)| < \varepsilon$),
2. vylepšení taktiky $\pi \to \pi'$ (I krok):
   - přizpůsob se novému ohodnocení stavů,
   - deterministické π: v každém stavu volí jedinou akci,
     platí-li $Q^\pi(s, \pi'(s)) \geq V^\pi(s)$ pro $\forall s$, π' je lepší nebo stejná jako π,
     zřejmě $\pi'(s) = \arg\max_a Q^\pi(s, a)$, volí aktuálně nejlepší akci,
   - stochastické π: volba akce je dána pstí,
     stejná logika pouze $Q^\pi(s, \pi'(s)) = \sum_a \pi'(s, a) Q^\pi(s, a)$,
     opět iterujeme tak dlouho, dokud dochází ke změně taktiky alespoň v jednom ze stavů.







$r = -1$
on all transitions

:: Vyhodnocení náhodné taktiky:

(hladová deterministická pouze ilustrací)

- $V(s) = \sum_a \pi(s,a) \sum_{s'} P^a_{ss'} \left[R^a_{ss'} + \gamma V^\pi(s') \right]$, $\pi(s,a) = 1/4, R^a_{ss'} = -1, P^a_{ss'} = 1, \gamma = 1$
- k=0: $\forall s V(s) = 0$, u koncových stavů vždy zůstává
- k=1: $V(1) = V(2) = \ldots = V(14) = \sum_a \pi(s,a) \sum_{s'} P^a_{ss'} \left[R^a_{ss'} + \gamma V^\pi(s') \right] = 4(1/4 * 1(-1 + 1 * 0)) = -1$
- k=2: $V(1) = 1/4(3 * 1 * (-1 + 1(-1)) + 1 * 1(-1 + 1 * 0)) = -7/4 = -1.75$
- k=3: $V(1) = 1/4(2 * 1 * (-1 + 1(-2)) + 1 * 1(-1 + 1(-1.75)) + 1 * 1(-1 + 1 * 0)) = -9.75/4 = -2.44$

Hodnotová iterace (value iteration – VI)

- je nutné vyhodnocovat stavy pro danou taktiku dokonale?
  - pozdní iterace zpravidla nemění taktiku a konzumují většinu času celého dynamického alg.,
  - **hodnotová iterace** – vyhodnocení je zastaveno už po jediném kroku,
  - častější změna taktiky, někdy rychlejší konvergence, neplatí ale obecně,
- z pohledu Bellmanovy rovnice dojde ke změně na přepisovací pravidlo

$$V^{\pi}(s) = \max_{a} \sum_{s'} P^{a}_{ss'} \left[R^{a}_{ss'} + \gamma V^{\pi}(s') \right]$$

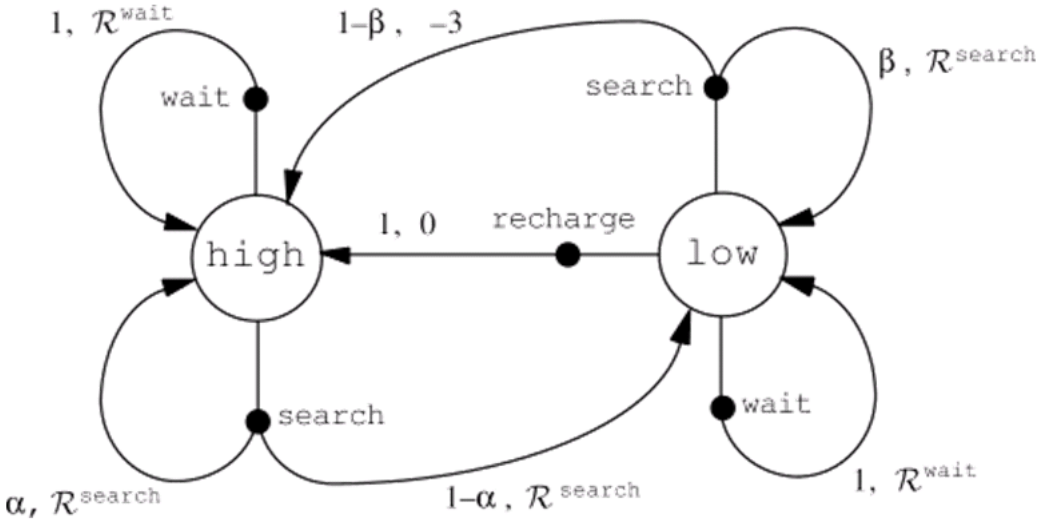
MDP – sběrný robot

:: Mobilní robot pro sběr plechovek

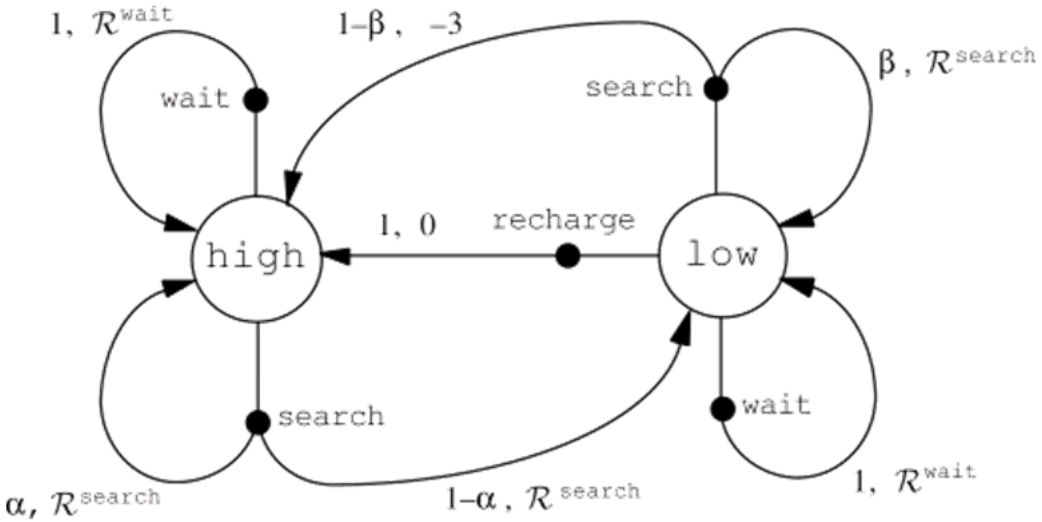
- má dva vnitřní stavy (baterie low a high),
- tři možné akce (aktivně sbírej, čekej na dobití, jeď se dobít sám),
- odměna je pozitivní při sebrání plechovky a velká negativní při vybití baterie uprostřed hledání.

:: **Logický cíl**: s minimem vnějších zásahů posbírat co nejvíce plechovek.

:: **Technický cíl**: vytvořit taktiku maximalizující dlouhodobou odměnu.



Sběrný robot – DP řešení



- Bellmanova rovnice: $V^\pi(s) = \sum_a \pi(s,a) \sum_{s'} P^a_{ss'} \left[R^a_{ss'} + \gamma V^\pi(s') \right]$
- Iterační rovnice pro konkrétní deterministické volby akcí (taktiky), high=h, wait=w, atd.:

$$\begin{aligned}
\pi(h,w) = 1: &\quad V(h) = Q(h,w) = R^w + \gamma V(h) \\
\pi(h,s) = 1: &\quad V(h) = Q(h,s) = R^s + \gamma \left[\alpha V(h) + (1-\alpha) V(l) \right] \\
\pi(l,r) = 1: &\quad V(l) = Q(l,r) = \gamma V(h) \\
\pi(l,w) = 1: &\quad V(l) = Q(l,w) = R^w + \gamma V(l) \\
\pi(l,s) = 1: &\quad V(l) = Q(l,s) = \beta R^s - 3(1-\beta) + \gamma \left[\beta V(l) + (1-\beta) V(h) \right]
\end{aligned}$$

Sběrný robot – DP řešení

:: **Parametry**: $\alpha = 0.95$, $\beta = 0.9$, $R^s = 2$, $R^w = 1$, $\gamma = 0.9$, $\varepsilon = 0.01$

:: **Metoda**: iterace taktiky (EI cyklus)

1. Zvol náhodně deterministickou taktiku: $\pi(low, wait) = \pi(high, wait) = 1$.
2. Nastav $V(low) = V(high) = 0$.
3. Použij příslušné iterační rovnice a počítej dokud se hodnoty V neustálí.
4. Dle ohodnocení V stanov nové otimální akce: $V(s) = \max_a Q^\pi(s, a)$, $\pi'(s) \approx \arg\max_a Q^\pi(s, a)$
5. Pokud nedojde ke změně taktiky skonči, jinak jdi na krok 2.

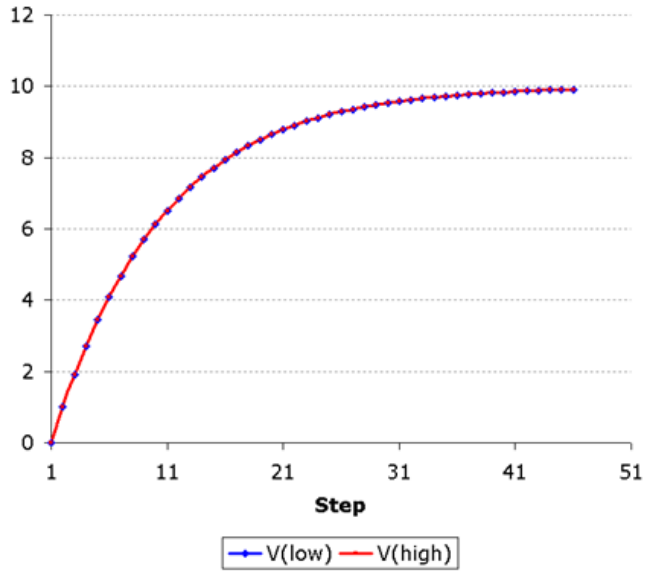
Inicializace:

$\pi(l, w) = \pi(h, w) = 1$

$V(h) = R^w + \gamma V(h)$, $V(l) = R^w + \gamma V(l)$

Vyhodnocení 1:

$V(h) = V(l) = 10$, 46 kroků



Vylepšení 1:

$\pi(l, s) = \pi(h, s) = 1$

$V(h) = R^s + \gamma[\alpha V(h) + (1 - \alpha)V(l)]$

$V(l) = \beta R^s - 3(1-\beta) + \gamma[\beta V(l) + (1-\beta)V(h)$

Vyhodnocení 2:

$V(h) = 19$, $V(l) = 16.8$, 52 kroků

Vylepšení 2:

$\pi(l, r) = \pi(h, s) = 1$

$V(h) = R^s + \gamma[\alpha V(h) + (1 - \alpha)V(l)]$

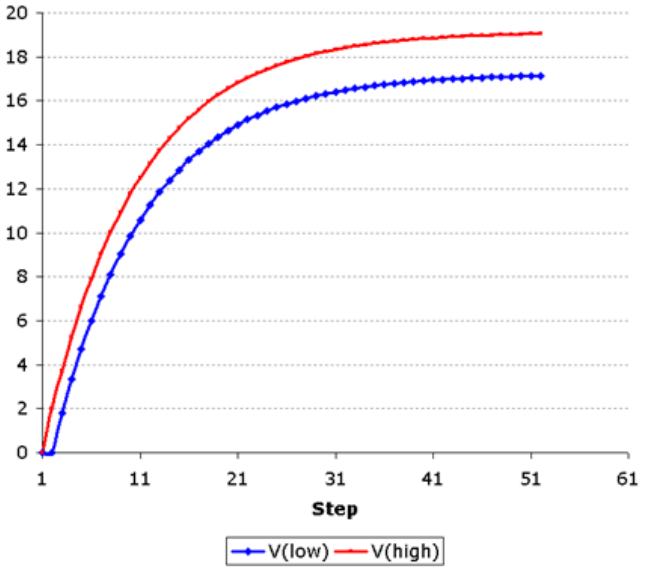
$V(l) = \gamma V(h)$

Vyhodnocení 3:

$V(h) = 19.1$, $V(l) = 17.1$, 52 kroků

Vylepšení 3:

$\pi(l, r) = \pi(h, s) = 1 \rightarrow$ STOP



SHRNUTÍ:

taktika: low $\rightarrow$ recharge, high $\rightarrow$ search

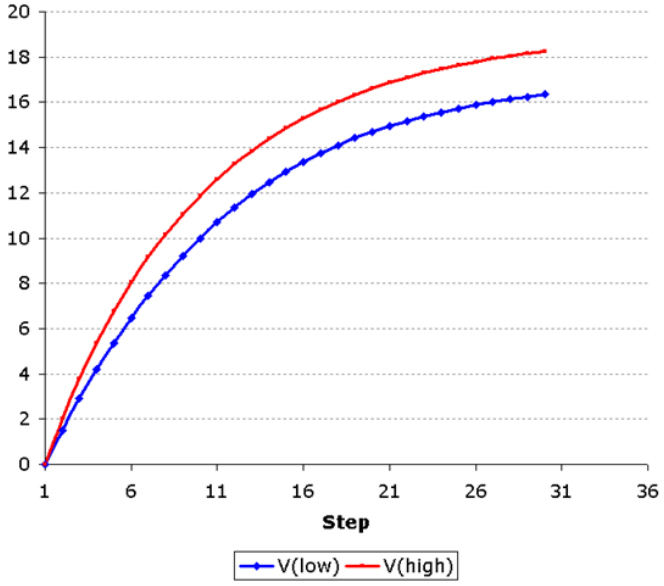
$V(h) = 19.1$, $V(l) = 17.1$, 150 iteračních kroků

Sběrný robot – DP řešení

:: **Parametry**: $\alpha = 0.95$, $\beta = 0.9$, $R^s = 2$, $R^w = 1$, $\gamma = 0.9$, $\varepsilon = 0.01$,

:: **Metoda**: hodnotová iterace

1. Nastav $V(low) = V(high) = 0$.
2. Dle ohodnocení V stanov nové otimální akce: $V(s) = max_a Q^\pi(s,a)$, $\pi'(s) \approx \arg\max_a Q^\pi(s,a)$.
3. Jednou aplikuj současné optimální akce a přepočítej hodnoty $V(s)$.
4. Pokud se hodnota žádného ze stavů nezměnila o více než ε skonči, jinak jdi na krok 2.



krok 0: $V(l) = V(h) = 0$, $\pi(l,s) = \pi(h,s) = 1$

krok 1: $V(l) = 1.5$, $V(h) = 2$, $\pi(l,s) = \pi(h,s) = 1$

krok 9: $V(l) = 9.2$, $V(h) = 11.1$, $\pi(l,r) = \pi(h,s) = 1$, změna taktiky

krok 52: $V(l) = 17.1$, $V(h) = 19.1$, $\pi(l,r) = \pi(h,s) = 1$, změny V menší než ε, STOP

SHRNUTÍ:

taktika: low $\rightarrow$ recharge, high $\rightarrow$ search

$V(h) = 19.1$, $V(l) = 17.1$, 52 iteračních kroků

Proč hodnotová iterace zaručeně konverguje?

- kontrakce $c(x)$
  - $d(c(x_1) - c(x_2)) \leq kd(x_1 - x_2)$, d je metrikou (vzdálenostní fcí), konstanta $k < 1$,
  - pevný bod b_c: $c(b_c) = b_c$, $c(c(\ldots c(x))) = b_c$,
  - každá kontrakce má právě jeden pevný bod,
  - příklad: $c(x) = \frac{x}{2}$, $d(x, y) = |x - y|$, $b_c = 0$,
- rovnice hodnotové iterace: $V_{i+1}(s) = \max_a \sum_{s'} P^a_{ss'} \left[R^a_{ss'} + \gamma V_i(s')\right]$
  - zapíšeme zjednodušeně jako $V_{i+1} \leftarrow BV_i$,
  - jako d použijeme max normu $||V|| = max_s|V(s)|$,
- výše definované B je vzhledem k $||\ ||$ kontrakcí (bez důkazu)
  - pro jakoukoli dvojici vektorů užitku stavů platí

    $$||BV_i - BV_i'|| \leq \gamma||V_i - V_i'|| \Rightarrow ||V_{i+1} - V_i|| \leq \gamma||V_i - V_{i-1}||,$$

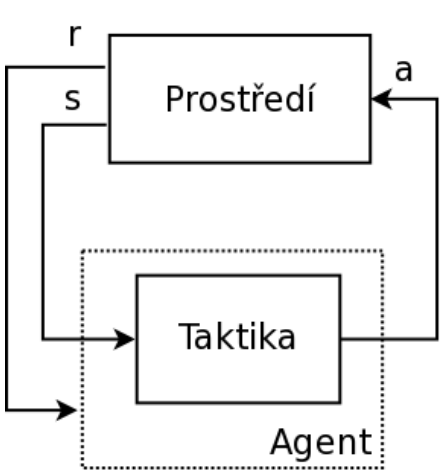
    * hodnotová iterace konverguje pro $\gamma < 1$,
  - pevným bodem je skutečný vektor užitku stavů V^*
    * $||BV_i - V^*|| \leq \gamma||V_i - V^*||$,
    * konverguje exponenciálně s γ.

Částečná pozorovatelnost

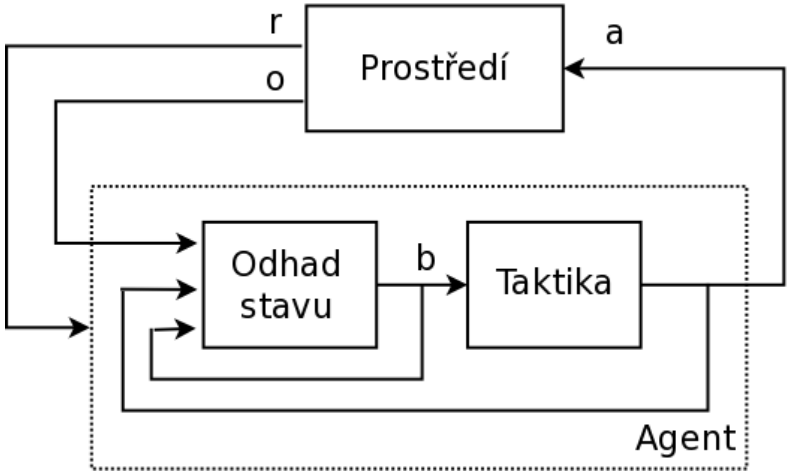
- MDPs pracují s předpokladem úplné pozorovatelnosti
  - stav je vždy známý,
  - velmi silný a často nerealistický předpoklad,
- částečně pozorovatelný Markovův rozhodovací proces
  - partially observable Markov decision process (POMDP),
  - zobecnění MDP, dynamika světa popsána jako u MDP,
  - na stav může usuzovat pouze z dílčích pozorování,
  - POMDP $= \{S, A, P, R, O, \Omega\}$,
    - O je množina pozorování,
    - Ω je senzorický model – definuje podmíněné psti pozorování, $\Omega^a_{s'o} = Pr\{o_{t+1} = o \mid s_{t+1} = s', a_t = a\}$
  - namísto stavu s vnitřně udržuje rozdělení psti přes možné stavy b (**belief**),
    - v neznámém stavu s (známe pouze $b(s)$) provedeme a a následně pozorujeme o $b'(s') = \eta\, \Omega^a_{s'o} \sum_{s \in S} P^a_{ss'} b(s)$,
    - η je normalizační konstanta volená tak, aby $\sum_{s' \in S} b'(s') = 1$.

Částečná pozorovatelnost

- důsledky částečné pozorovatelnosti
  - nemá smysl hovořit o taktice $\pi : S \rightarrow A$, hledá taktiku $\pi : B \rightarrow A$,
  - obvykle výpočetně nezvladatelné, pouze přibližná řešení
    - pro n stavů, je b n-dimenzionální reálný vektor, PSPACE-těžký, horší než NP.



a) MDP

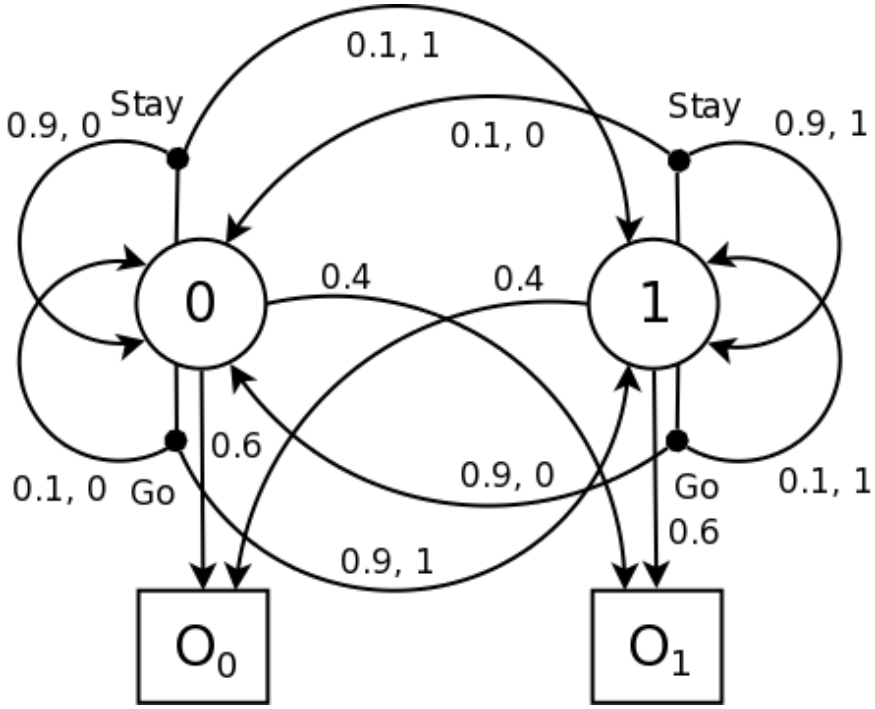


b) POMDP

Částečná pozorovatelnost – příklad

:: $S = \{0, 1\}$, $A = \{Stay, Go\}$, $P(s_{t+1} = x | s_t = x, Stay) = .9$, $P(s_{t+1} = x | s_t = x, Go) = .1$,
. $O = \{o_0, o_1\}$, $Pr(o_0|0) = .6$, $Pr(o_1|1) = .6$, $R(0) = 0$, $R(1) = 1$, $\gamma = 1$,

:: Cíl: stanovit $V^*(b)$ (hlavní krok pro stanovení (deterministické) taktiky $a = \pi^*(b)$)



Částečná pozorovatelnost – příklad

- b prostor je jednodimenizonální, hledáme reálnou fci jedné proměnné,
- předpokládáme, že v blízkých bodech b prostoru bude
  - velmi podobný užitek,
  - shodná taktika,
- aplikace taktiky odpovídá podmíněnému plánu
  - akce budou záviset na budoucích pozorováních,
  - příklad plánu délky 2: $[Stay, \text{if } O = o_0 \text{ then } Go \text{ else } Stay]$,
- užitek plánu p počínajícího ve stavu s nechť je $\alpha_p(s)$,
  - pak stejný plán provedený z b má užitek

$$\sum_s b(s)\alpha_p(s) = b \cdot \alpha_p$$

  - je lineární funkcí b (nadrovinou pro složité prostory),
  - optimální taktika volí plán s nejvyšším středním užitkem

$$V(b) = V^{\pi^*}(b) = \max_p b \cdot \alpha_p$$

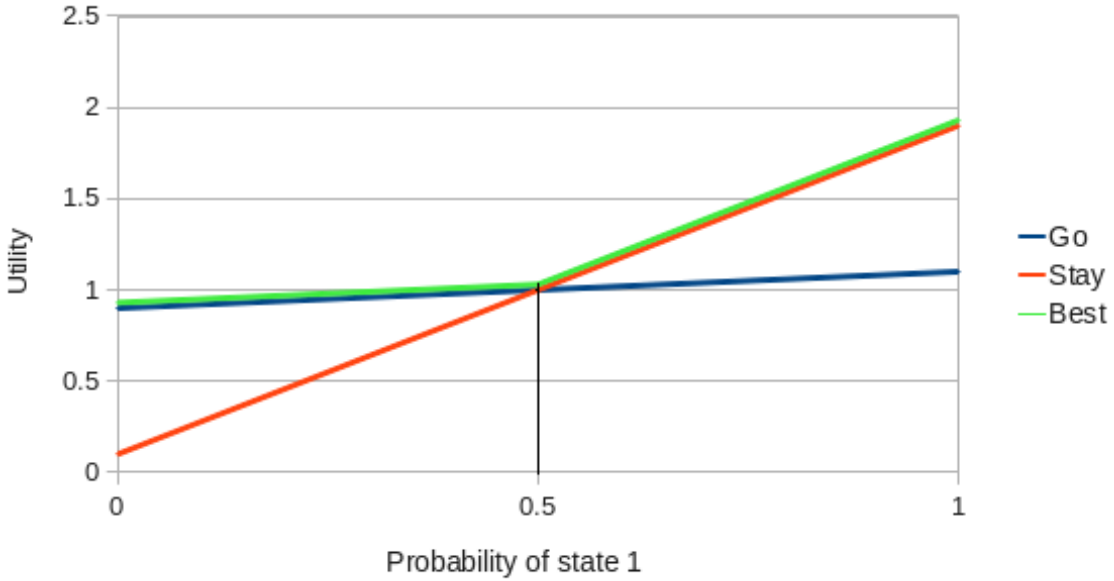
  - $V(b)$ je po částech lineární funkcí b.

Částečná pozorovatelnost – příklad

:: pro plány délky 1 (existují 2)

$$
\begin{aligned}
\alpha_{[Stay]}(0) &= R(0) + \gamma(.9R(0) + .1R(1)) &= 0.1 \\
\alpha_{[Stay]}(1) &= R(1) + \gamma(.9R(1) + .1R(0)) &= 1.9 \\
\alpha_{[Go]}(0) &= R(0) + \gamma(.9R(1) + .1R(0)) &= 0.9 \\
\alpha_{[Go]}(1) &= R(0) + \gamma(.9R(0) + .1R(1)) &= 1.1
\end{aligned}
$$

$$
\alpha_{[Stay]}(b(1) = 0.3) = .7\alpha_{[Stay]}(0) + .3\alpha_{[Stay]}(1) = 0.64
$$

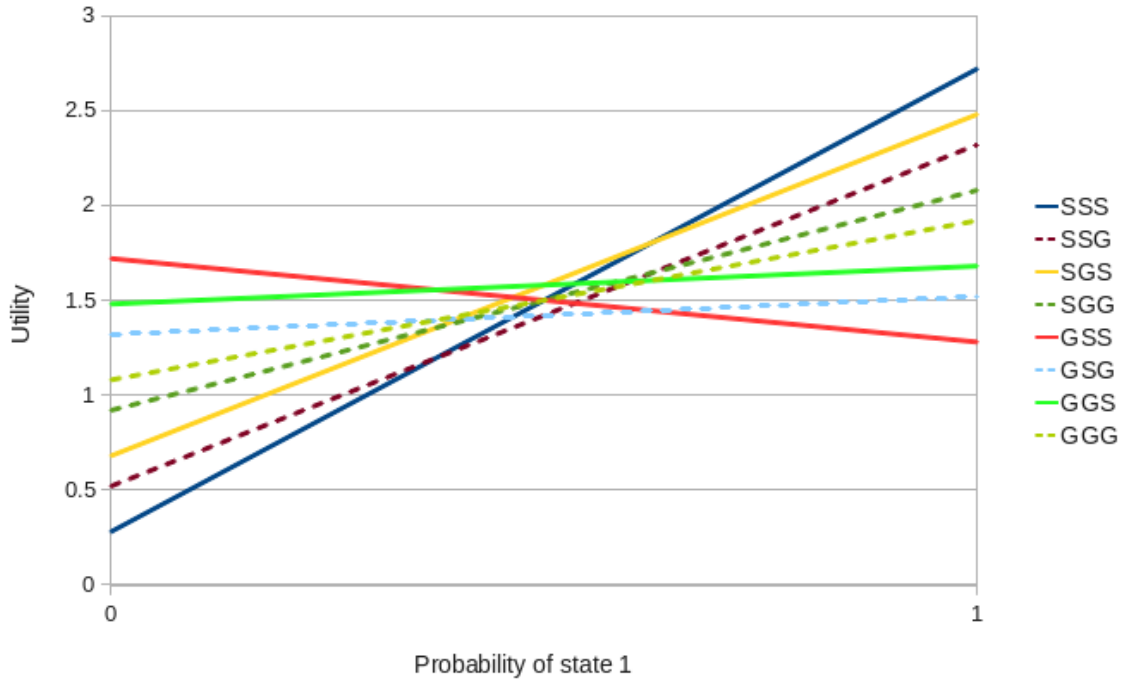


Částečná pozorovatelnost – příklad

:: pro plány délky 2 (existuje jich 8, 4 čárkované čistě horší než jiné)

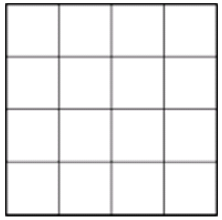
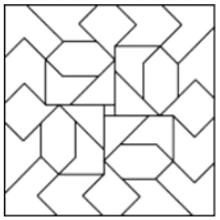
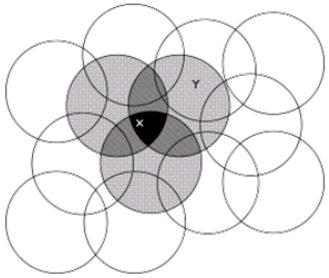
$[Stay, \text{if } O = o_0 \text{ then } Go \text{ else } Stay]$ kódován jako $[SGS]$

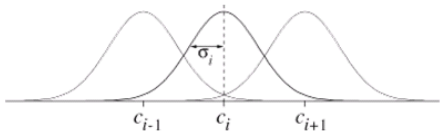
$$
\begin{aligned}
\alpha_{[SSS]}(0) &= R(0) + \gamma(.9\alpha_{[S]}(0) + .1\alpha_{[S]}(1)) &&= 0.28\\
\alpha_{[SSS]}(1) &= R(1) + \gamma(.9\alpha_{[S]}(1) + .1\alpha_{[S]}(0)) &&= 2.72\\
\alpha_{[SGS]}(0) &= R(0) + \gamma(.9(.6\alpha_{[G]}(0) + .4\alpha_{[S]}(0)) + .1(0.4\alpha_{[G]}(1) + .6\alpha_{[S]}(1)) &&= 0.68\\
\alpha_{[SGS]}(1) &= R(1) + \gamma(.9(.4\alpha_{[G]}(1) + .6\alpha_{[S]}(1)) + .1(0.6\alpha_{[G]}(0) + .4\alpha_{[S]}(0)) &&= 2.48
\end{aligned}
$$



Generalizace a aproximace

- dosud triviální úlohy s malým počtem stavů a akcí,
- pokud je stavů hodně je nutná aproximace hodnotových fcí
  - k tomu lze využít učení z příkladů,
  - generalizace předpokládá spojité a "rozumné" hodnotové funkce,
  - stavy jsou charakterizovány vektorem parametrů/atributů ϕ_s,



  - zvolíme typ aproximační funkce a snažíme se optimalizovat vektor jejích parametrů $\theta(t)$,
  - lineární aproximace: $V_t(s) = \theta_t^T \phi_s = \sum_i \theta_t(i)\phi_s(i)$,
  - nelineární optimalizace neuronovou sítí,
  - minimalizuje chybovou funkci: $MSE(\theta_t) = \sum_s P_s \left[V_t^\pi(s) - V_t(s)\right]^2$,
    (P_s distribuce váhy přes stavy, $V_t^\pi(s)$ skutečná hodnota stavu, $V_t(s)$ její aproximace,)
  - regrese, gradientní optimalizace, zpětná propagace apod.

Shrnutí

- MDPs zobecňují prohledávání deterministických stavových prostorů na stochastické,
  - cenou je výpočetní náročnost,
- řešením problému je nalezení taktiky, která každému stavu přiřazuje optimální akci
  - taktika může být stochastická,
  - základní přístupy jsou iterace taktiky a hodnotová iterace,
  - další možnosti jsou modifikované iterační přístupy, případně asynchronní,
- příbuzné techniky
  - POMDP pro částečně pozorovatelná prostředí,
  - RL pro neznámé modely prostředí,
- aplikace
  - obecně agentní technologie,
  - v robotice – řízení robota, plánování cesty,
  - v telekomunikacích – optimalizace sítí,
  - při hraní her.

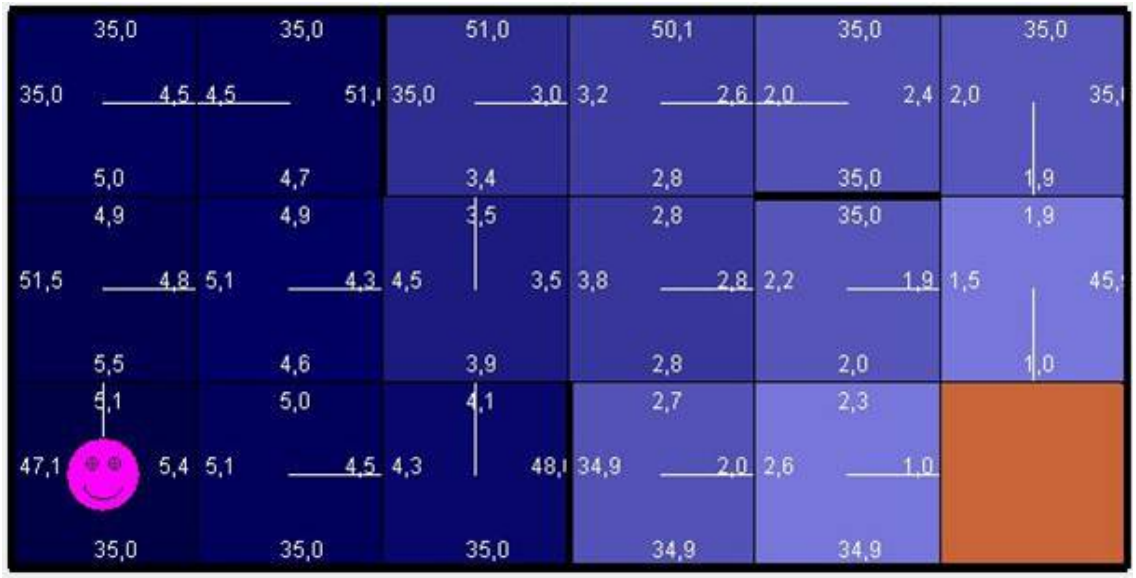
Doporučené doplňky – zdroje přednášky

:: Četba

- Russell, Norvig: **AI: A Modern Approach**, Making Complex Decisions
  - kapitola 17,
  - kniha online on Google books: http://books.google.com/books?id=8jZBksh-bUMC,
- Richard S. Sutton, Andrew G. Barto: **Reinforcement Learning: An Introduction**, MIT Press, Cambridge, 1998.
  - http://www.cs.ualberta.ca/~sutton/book/the-book.html.

Ukázka

- RL simulátor
  - hledání optimální cesty k cíli bludištěm
  - implementace v Javě
  - http://www.cs.cmu.edu/~awm/rlsim/



©Kelkar, Mehta: Robotics Institute, Carnegie Mellon University