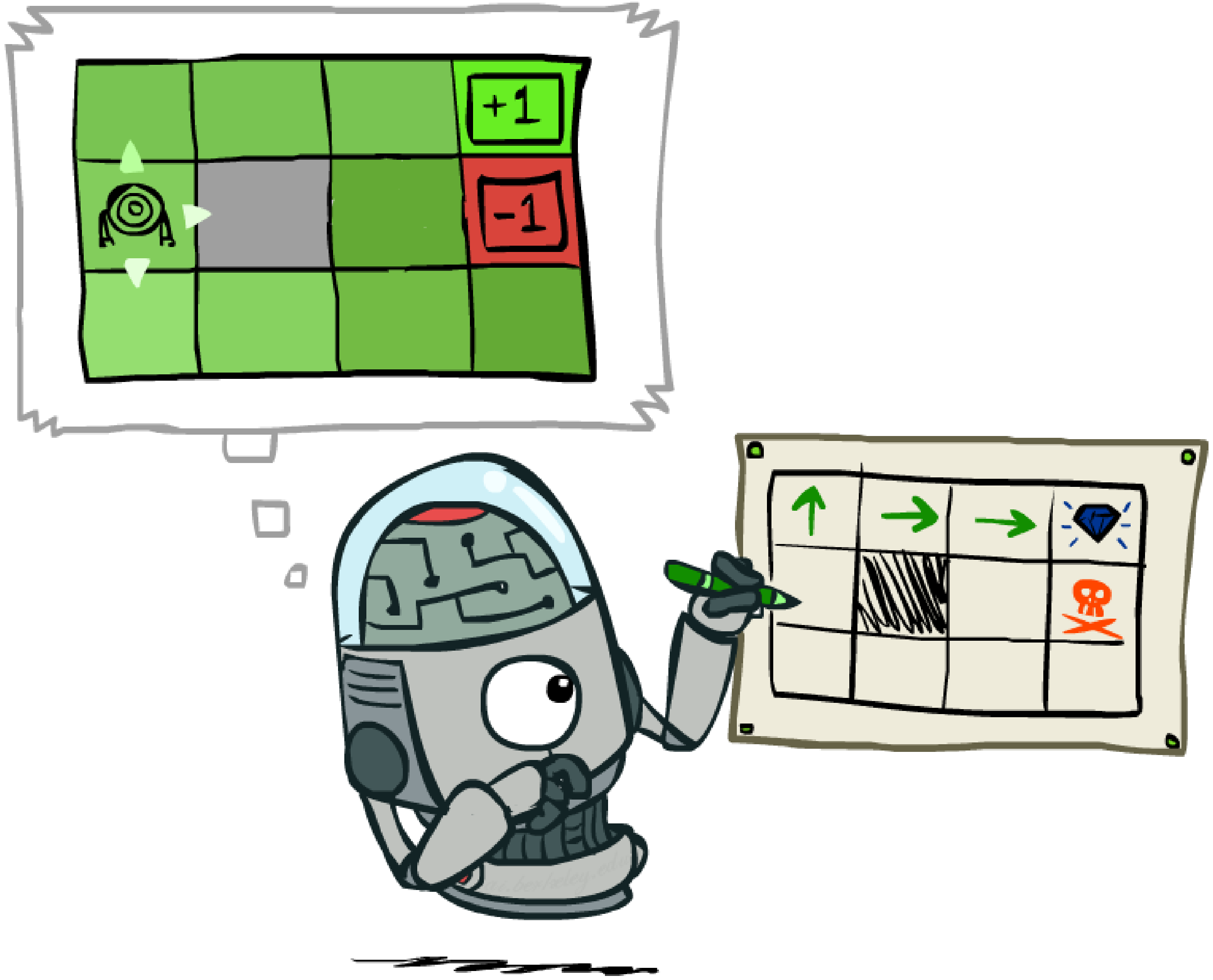
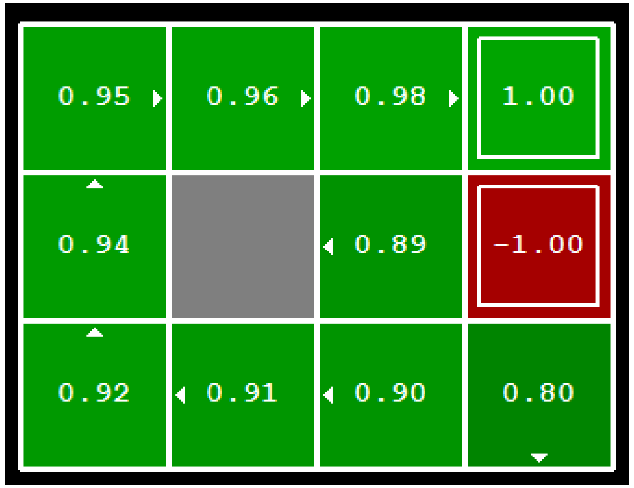
Ektrakcija Politike (*Policy Extraction*)



Određivanje Akcija iz Vrednosti

- Prepostavimo da imamo optimalne vrednosti V*(s)

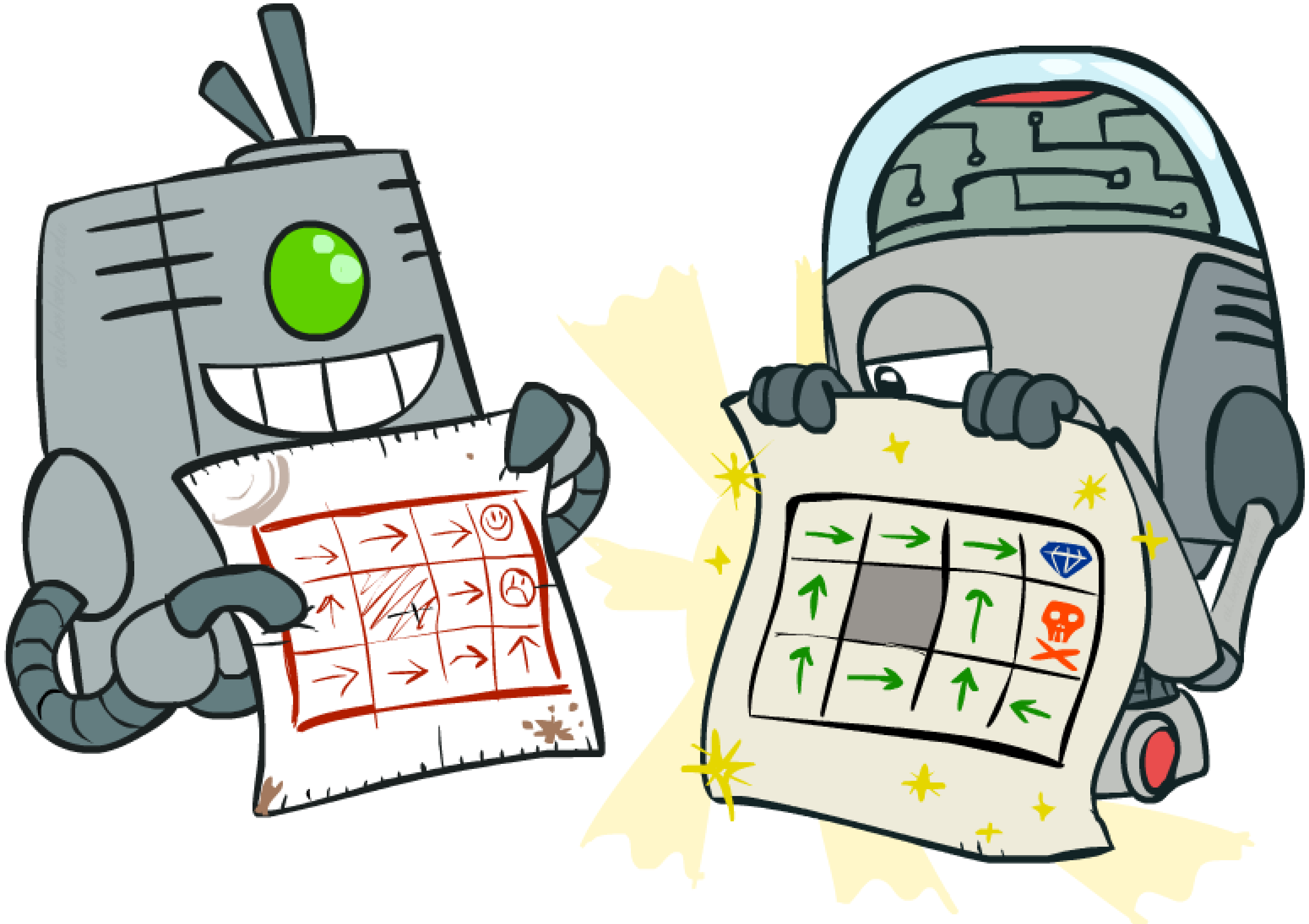


- Koje akcije treba da radimo?
  - Nije očigledno!
- Treba da uradimo mini-expectimax (jedan korak)

$$\pi^*(s) = \arg\max_a \sum_{s'} T(s,a,s')[R(s,a,s') + \gamma V^*(s')]$$

- Ovo se zove ekstrakcija politike, jer nam daje politiku koja je implicitno data vrednostima.

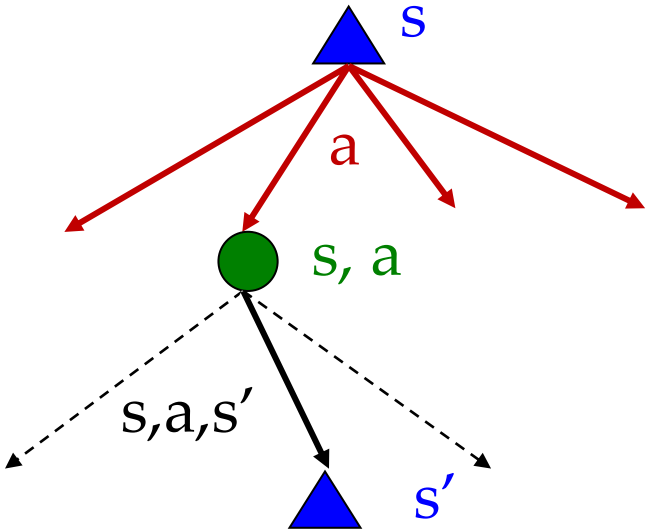
Metodi Vezani za Politike (*Policy Methods*)



Problem sa Iteriranjem Vrednosti

- Algoritam iterira Belmanove promene:

$$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma V_k(s')\right]$$

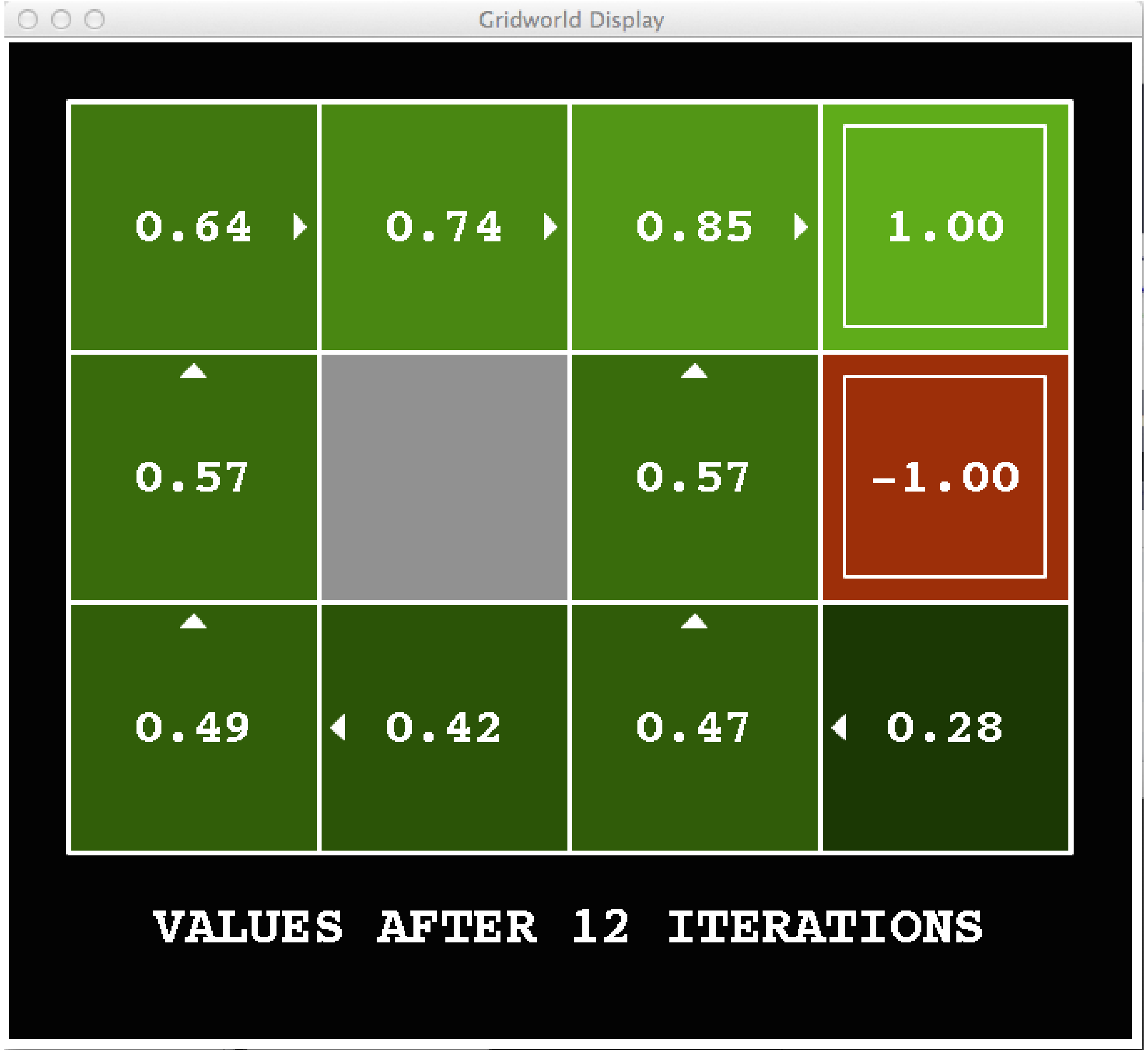


- Problem 1: Spor je – $O(S^2A)$ po iteraciji

- Problem 2: Max po akcijama dat gore se retko menja tj. vrednost će se možda promeniti, ali akcija koja ga je dala se retko menja

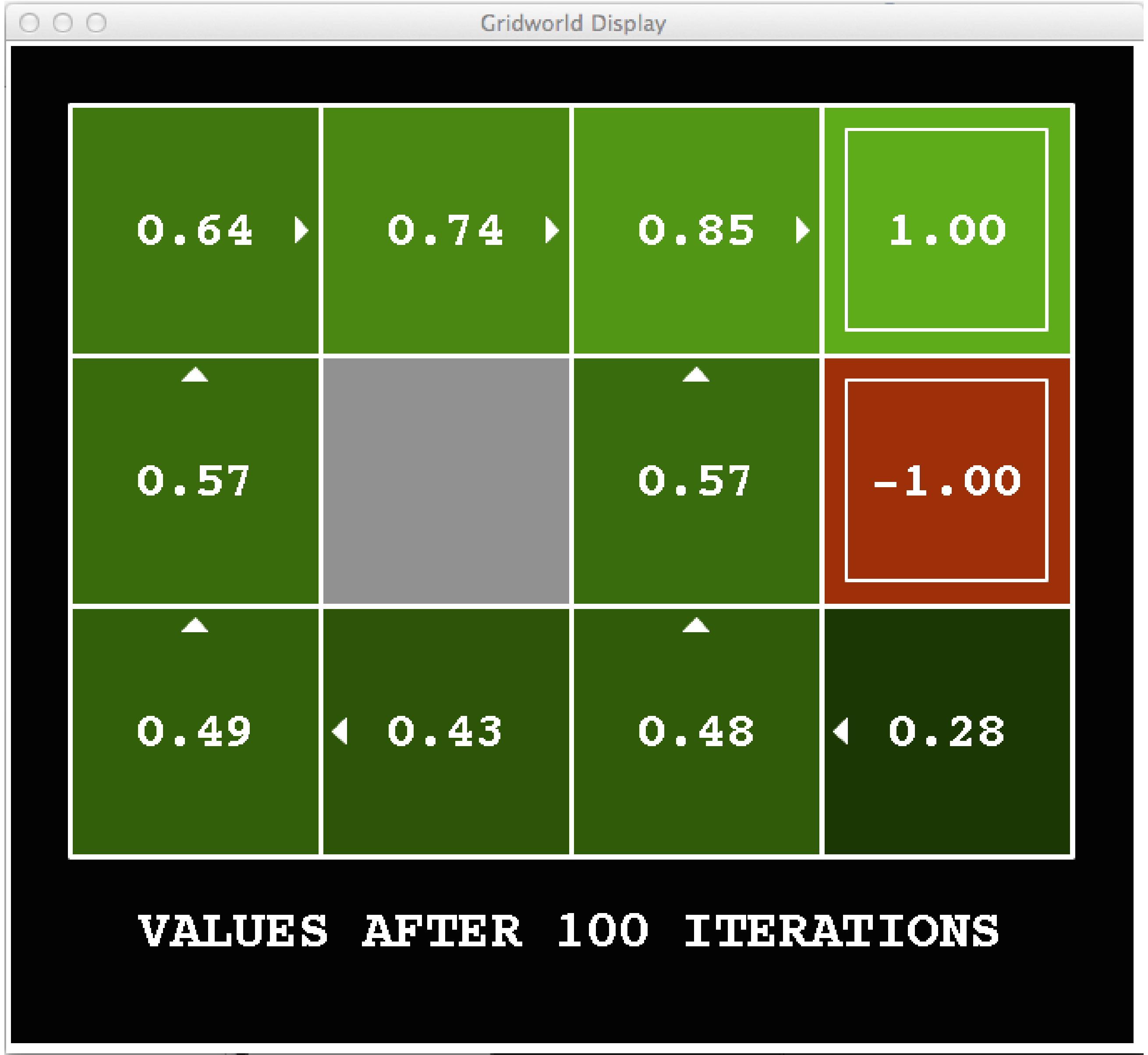
- Problem 3: U vezi sa P2, politika konvergira mnogo pre vrednosti

k=12



Šum = 0.2
Zanemarivanje = 0.9
Nagrada postojanja = 0

k=100



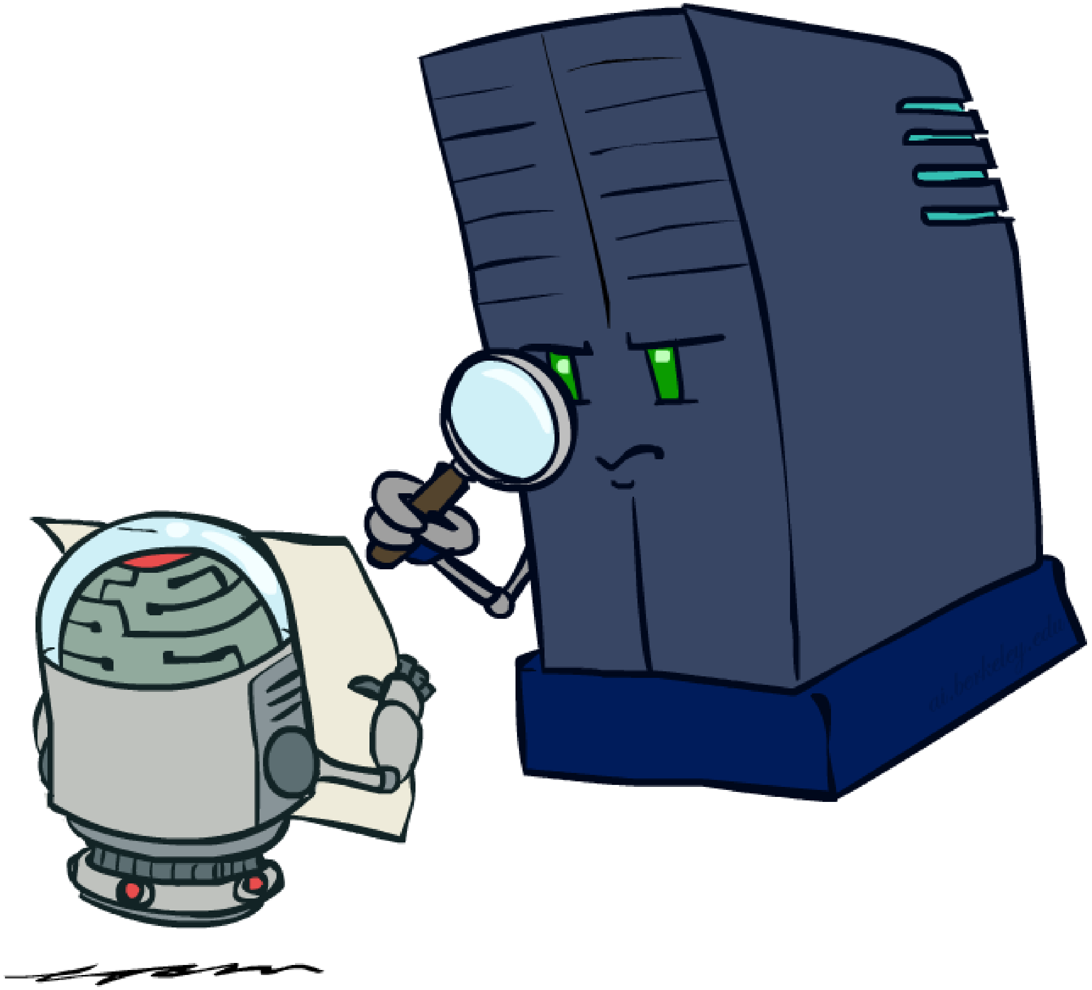
Šum = 0.2
Zanemarivanje = 0.9
Nagrada postojanja = 0

Iteriranje Politike (*Policy Iteration*)

- Alternativna tehnika za dobijanje optimalnih vrednosti:
  - Korak 1: Evaluacija politike: uzmemo neku fiksnu politiku i izračunavamo vrednosti stanja pomoću nje (ovo neće biti optimalne vrednosti!) do konvergencije
  - Korak 2: Poboljšanje politike: menjamo politiku tako što u svakom stanju biramo onu akciju koja će nam maksimizovati buduću nagradu (uz zanemarivanje) – za izračunavanje nagrade koristimo vrednosti koje smo dobili u koraku 1.
  - Ponavljamo dok politika ne konvergira

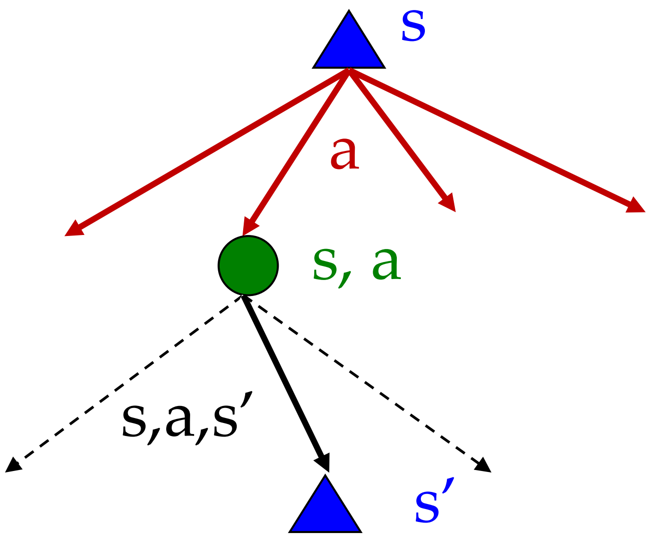
- Ovo je iteriranje politike
  - Takođe dobijamo optimalnu politiku!
  - Često konvergira brže od Iteriranja Vrednosti

Evaluacija Politike

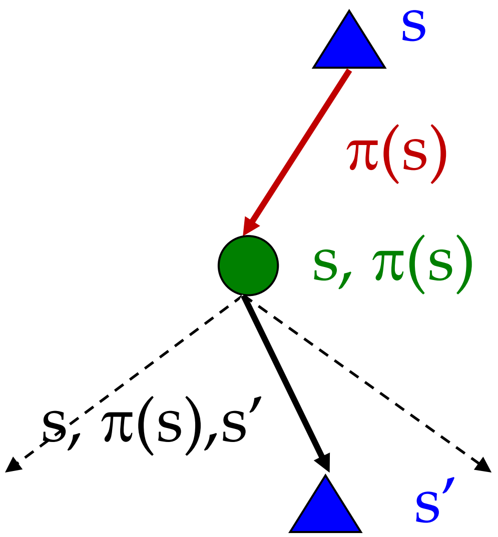


Fiksirane Politike

Radi optimalnu akciju



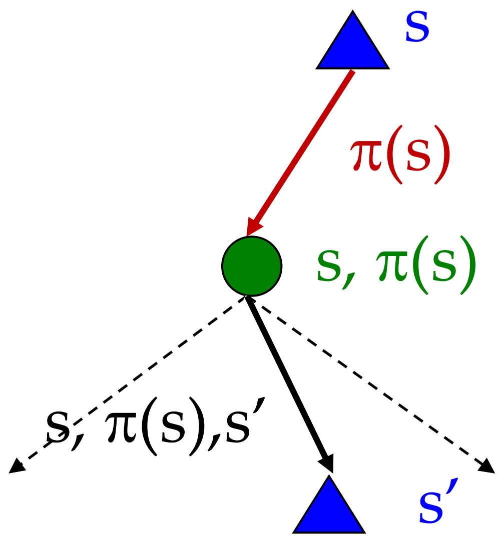
Radi šta ti π kaže



- Algoritmi do sada traže max po svim akcijama za dato stanje
- Ako fiksiramo politiku π(s), stablo je jednostavnije – samo jedna akcija po stanju

Korisnosti za Fiksiranu Politiku

- Još jedna vrsta izračunavanja: izračunati vrednost (korisnost) za s, ali za fiksiranu politiku (koja ne mora biti optimalna)



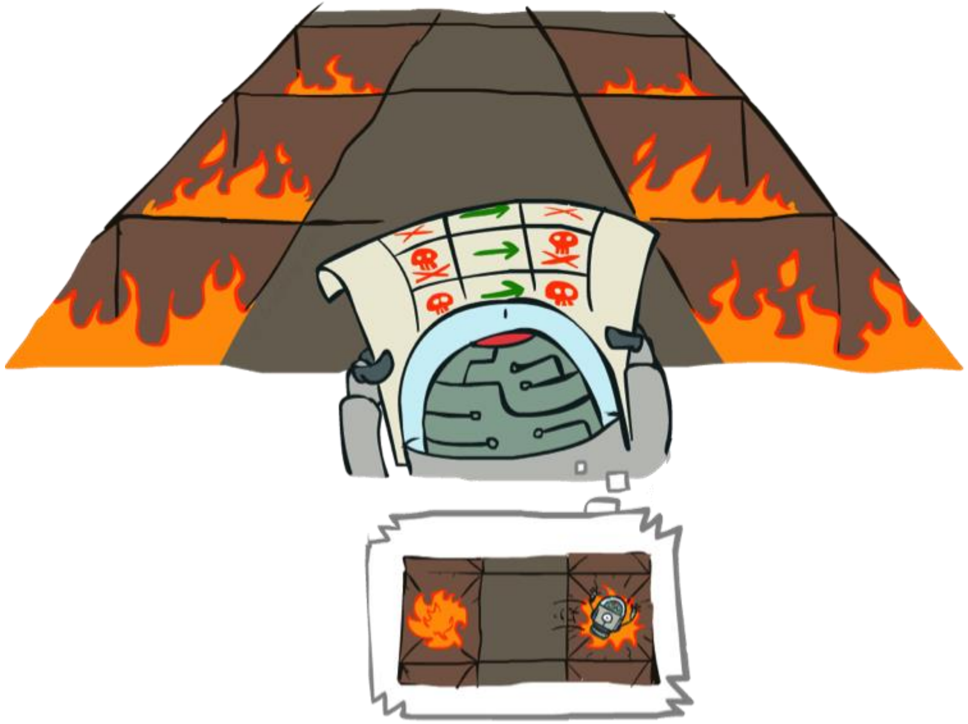
- Vrednost za stanje s, za fiksnu politiku π definišemo sa:
  - $V^\pi(s)$ = sumu očekivanih nagrada (uz zanemarivanje) ako krenemo iz s i uvek pratimo π

- Rekurzivna relacija („pogled" jedan korak napred / Belmanova jednakost):

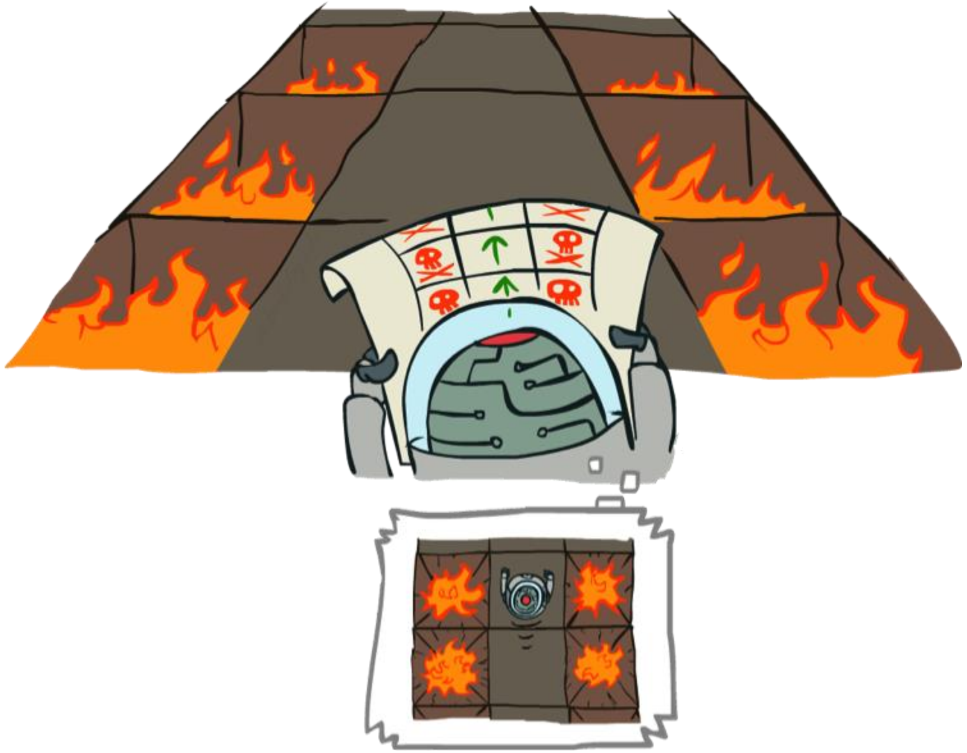
$$V^\pi(s) = \sum_{s'} T(s,\pi(s),s')[R(s,\pi(s),s') + \gamma V^\pi(s')]$$

Primer: Evaluacija Politike

Always Go Right



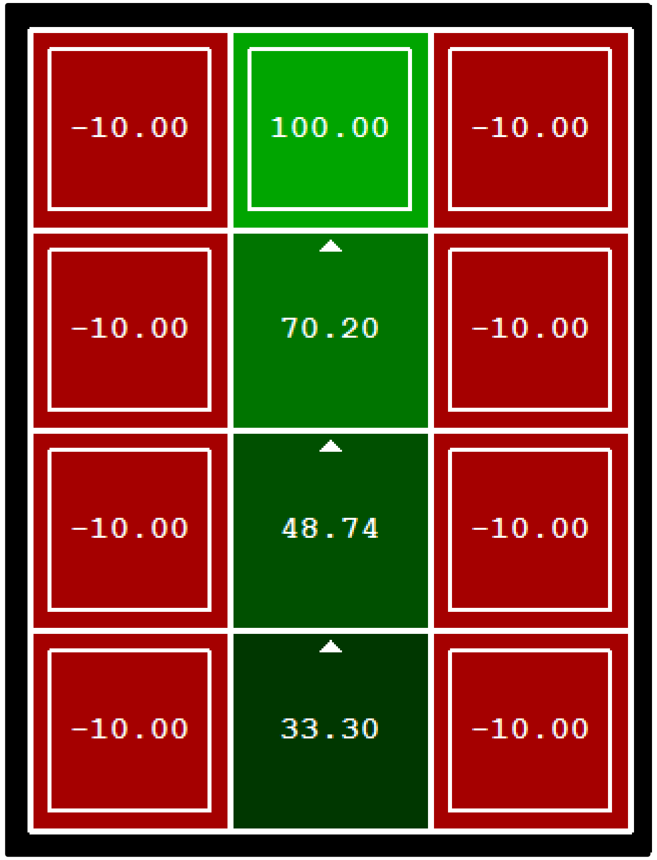
Always Go Forward



Primer: Evaluacija Politike

Always Go Right

Always Go Forward



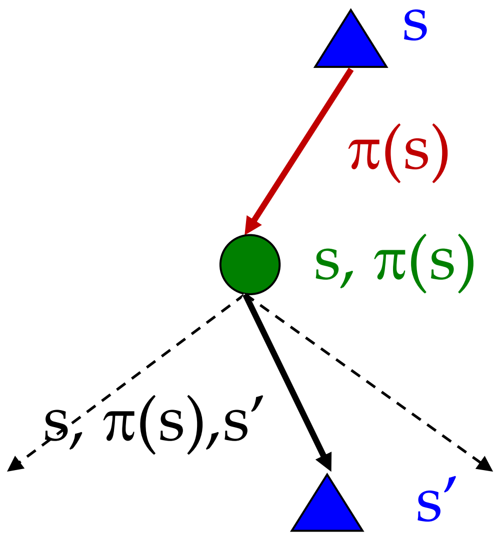
Evaluacija Politike

- Kako izračunavamo vrednosti V za fiksiranu politiku π?

- Ideja 1: Koristimo Belmanovu jednakost da radimo promene (kao iteriranje vrednosti)

$$V_0^\pi(s) = 0$$

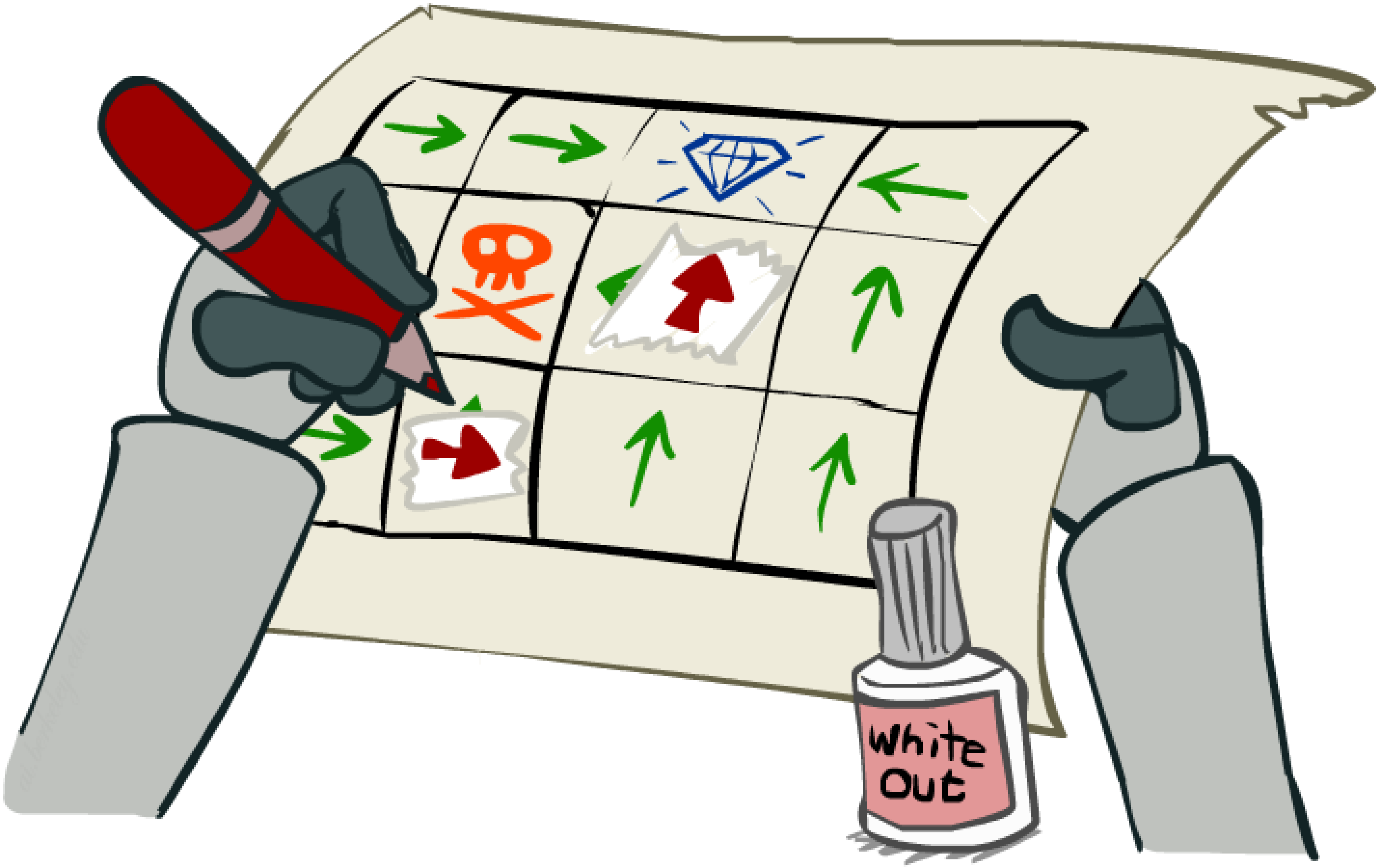
$$V_{k+1}^\pi(s) \leftarrow \sum_{s'} T(s, \pi(s), s')[R(s, \pi(s), s') + \gamma V_k^\pi(s')]$$



- Efikasnost: $O(S^2)$ po iteraciji

- Ideja 2: Bez max ispred suma, Belmanove jednačine za svako stanje formiraju sistem linearnih jednačina
  - Koristimo neki od numeričkih metoda

Iteriranje Politike



Iteriranje Politike

- Evaluacija: Za fiksiranu politiku π, izračunati vrednosti pomoću evaluacije politike:
  - Iterirati do konvergencije:

$$V_{k+1}^{\pi_i}(s) \leftarrow \sum_{s'} T(s, \pi_i(s), s') \left[R(s, \pi_i(s), s') + \gamma V_k^{\pi_i}(s')\right]$$

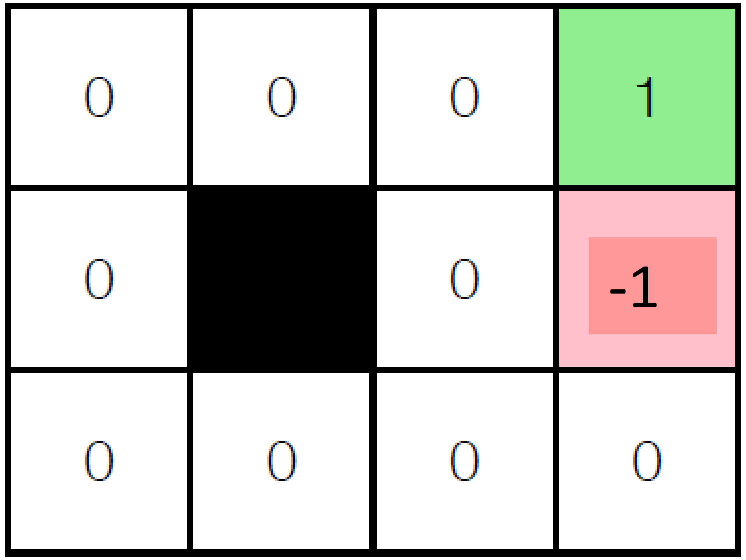
- Poboljšanje: Za fiksirane vrednosti (iz evaluacije), popravljamo politiku
  - „Pogled jedan korak unapred“:

$$\pi_{i+1}(s) = \arg\max_a \sum_{s'} T(s, a, s') \left[R(s, a, s') + \gamma V^{\pi_i}(s')\right]$$

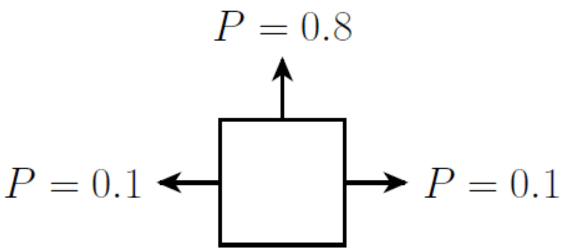
Evaluacija Politike - Primer

- Počinjemo sa politikom koja za svako stanje kaže da se ide na Sever. Zanemarivanje je 0.9. Nagrade su date na slici.

Running policy iteration with $\gamma = 0.9$, initialized with policy $\pi(s) = \text{North}$



Original reward function

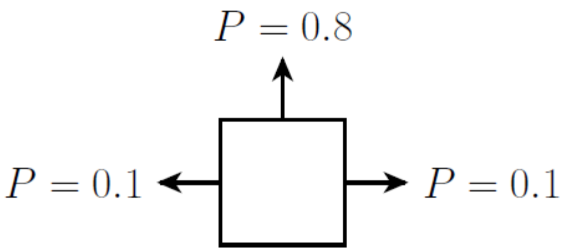


Evaluacija Politike - Primer

- Rezultati evaluacije politike „uvek idi na Sever“ pomoću Gausove eliminacije.

Running policy iteration with $\gamma = 0.9$, initialized with policy $\pi(s) = \text{North}$

0.168	0.375	0.442	1
0.147		0.250	-1
0.005	0.059	0.178	-0.08



Original reward function

Iteriranje Politike - Primer

- Posmatramo stanje koje ima vrednost 0.442. Označićemo ga sa *s*.
- Nova politika za to stanje biće najbolja akcija po Belmanovoj jednačini.

Q(s,east)=0.8*(0.9*1)+0.1*0.9*0.250+0.1*0.9*0.442

Q(s,west)=0.8*(0.9*0.375)+0.1*0.9*0.250+0.1*0.9*0.442

Q(s,north)=0.8*(0.9*0.442)+0.1*0.9*1+0.1*0.9*3.75

Q(s,south)=0.8*(0.9*0.250)+0.1*0.9*1+0.1*0.9*3.75

Q(s,east)=1.097

Q(s,west)=0.332

Q(s,north)=0.745

Q(s,south)=0.607

Vidimo da je Q(s,east) najveća vrednost.
To znači da sad popravljamo trenutnu politiku π(s,north) i menjamo je na π(s,east).
Nastavljamo ovaj proces za svako stanje. Dobijamo novu politiku. Onda radimo njenu evaluaciju, pa opet poravljanje itd. do konvergencije.

0.168	0.375	0.442	1
0.147		0.250	-1
0.005	0.059	0.178	-0.08

Poređenje

- Oba algoritma izračunavaju istu stvar (optimalne vrednosti za stanja)

- Kod iteriranja vrednosti:
  - Svaka iteracija osvežava vrednosti i (implicitno) politiku
  - Ne pratimo politiku tokom algoritma, ali je uvek možemo dobiti tako što biramo max akciju za vrednosti (ekstrakcija politike)

- Kod iteriranja politike:
  - Treba nam nekoliko koraka da bi izračunali vrednosti za politiku (svaki korak je brz jer imamo samo jednu akciju za svako stanje)
  - Nakon što evaluiramo politiku, poravljamo je na osnovu vrednosti koje imamo (ovo je sporo)
  - Dobićemo bolju politiku od prethodne (ili ako nema promene završavamo)

Rrezime: MDP Algoritmi

- Hoćemo da….
  - Izračunamo optimalne vrednosti: koristimo iteriranje vrednosti ili politike
  - Izračunamo vrednosti za fiksiranu politiku: koristimo evaluaciju politike
  - Dobijemo politiku iz vrednosti: koristimo ekstrakciju politike
- Svi ovi algoritmi izgledaju isto!
  - Suštinski jesu – svi su varijacija Belmanovih promena
  - Svi koriste jedan korak expectimax algoritma
  - Razlikuju se samo po tome da li koristimo fiksiranu politku ili tražimo max po akcijama