

Problemi planiranja i primene

Milan Banković¹

¹Matematički fakultet, Univerzitet u Beogradu

27. maj 2025.

Overview

- 1 Uvod
- 2 Klasično planiranje
- 3 Rešavanje problema klasičnog planiranja
- 4 Primeri i primene

Uvod

O automatskom planiranju

Automatsko planiranje je oblast veštačke inteligencije koja se bavi pronalaženjem **plana** za rešavanje nekog problema, u vidu sekvence **akcija** kojom se postiže zadati **cilj**.

Uvod

Ključni pojmovi

- **Stanje** okruženja u okviru kog se plan realizuje
 - okruženje može biti **statičko** ili **dinamičko**
 - **početno stanje**: može biti **deterministički** ili **nedeterministički** zadato
- **Akcije** koje primenjuje **agent** kojim se stanje menja
 - imaju svoje **preduslove** i **efekte**
 - efekti mogu biti **deterministički**, ili **nedeterministički**
 - mogu biti **trenutne** ili **sa trajanjem**
- **Cilj**: uslov koji treba da zadovoljava **završno stanje**
- **Plan**: sekvenca akcija koja transformiše početno stanje u završno stanje koje zadovoljava zadati cilj

Uvod

Vrste planiranja

- **Klasično planiranje**: statičko okruženje, početno stanje deterministički određeno, akcije trenutne i determinističke
- **Temporalno planiranje**: akcije imaju trajanje, preduslove i završne uslove
- **Probabilističko planiranje**: početno stanje i efekti akcija nisu deterministički određeni (umesto plana, traži se **politika**)
- **Kontinualno planiranje**: dinamičko okruženje, zahteva mogućnost **replaniranja** u hodu
- **Multi-agentsko planiranje**: više agenata istovremeno primenjuje akcije za ostvarivanje zadatog cilja
- ...

Uvod

O čemu je ovo izlaganje?

U ovom izlaganju se bavimo isključivo **klasičnim planiranjem**.

Overview

- 1 Uvod
- 2 Klasično planiranje**
- 3 Rešavanje problema klasičnog planiranja
- 4 Primeri i primene

Klasično planiranje

Definicija

Problem (klasičnog) planiranja je uređena četvorka $(\mathcal{F}, S_0, \mathcal{A}, G)$, gde je:

- $\mathcal{F}$ konačan skup **činjenica** o trenutnom stanju
 - iskazni atomi, ili
 - predikati primenjeni na objekte
- **stanja** su podskupovi skupa $\mathcal{F}$
- $S_0 \subseteq \mathcal{F}$ je **početno stanje**
- $\mathcal{A}$ skup **akcija**, pri čemu za svako $a \in \mathcal{A}$ imamo:
 - $c(a)$ - preduslov (u terminima činjenica iz $\mathcal{F}$)
 - $e(a)$ - efekat (u terminima promene stanja, tj. koje činjenice se dodaju, a koje uklanjaju iz stanja)
- G - **cilj**, izražen u terminima činjenica iz $\mathcal{F}$

Klasično planiranje

Definicija (nastavak)

Rešenje je **plan** $P = \{a_1, \dots, a_n\}$ takav da postoji niz stanja $S_0, \dots, S_n$, gde je S_0 početno stanje, takav da važi:

- za svako a_i , preduslov $c(a_i)$ važi u stanju S_{i-1}
- stanje S_i se dobija primenom efekta $e(a_i)$ na stanje S_{i-1}
- stanje S_n zadovoljava uslov cilja G

Broj akcija n u planu P se naziva **dužina plana**.

Obično je cilj pronaći plan najmanje moguće dužine.

Klasično planiranje

Primer

Majmun se nalazi na lokaciji A želi da dohvati banane koje vise na lokaciji C . Da bi mogao da ih dohvati, mora se popeti na kutiju postavljenu ispod banana. Kutija se trenutno nalazi na lokaciji B . Majmun može da se kreće između lokacija, da gura kutiju sa lokacije na lokaciju, da se penje i spušta sa kutije, kao i da uzme banane. Osmisliti plan za dohvatanje banana.

Klasično planiranje

Primer (nastavak)

- Početno stanje: $\{ \text{monkey_at}(A), \text{box_at}(B), \text{bananas_at}(C) \}$
- Moguće akcije:
 - $\text{move_monkey}(\text{from}, \text{to})$
 - preduslov $\text{monkey_at}(\text{from}), \neg \text{on_box}$
 - efekat $\neg \text{monkey_at}(\text{from}), +\text{monkey_at}(\text{to})$
 - $\text{move_box}(\text{from}, \text{to})$
 - preduslov $\text{monkey_at}(\text{from}), \text{box_at}(\text{from}), \neg \text{on_box}$
 - efekat $\neg \text{monkey_at}(\text{from}), \neg \text{box_at}(\text{from}), +\text{monkey_at}(\text{to}), +\text{box_at}(\text{to})$
 - $\text{climb_up}(\text{loc})$
 - preduslov $\text{monkey_at}(\text{loc}), \text{box_at}(\text{loc}), \neg \text{on_box}$
 - efekat $+\text{on_box}$
 - $\text{take_bananas}(\text{loc})$
 - preduslov $\text{monkey_at}(\text{loc}), \text{box_at}(\text{loc}), \text{on_box}, \text{bananas_at}(\text{loc}), \neg \text{have_bananas}$
 - efekat $+\text{have_bananas}$
- Cilj: have_bananas mora biti u završnom stanju

Klasično planiranje

Primer (nastavak)

Rešenje je sledeći plan:

```
move_monkey(A, B)  
move_box(B, C)  
climb_up(C)  
take_bananas(C)
```

Overview

- 1 Uvod
- 2 Klasično planiranje
- 3 Rešavanje problema klasičnog planiranja**
- 4 Primeri i primene

Rešavanje problema klasičnog planiranja

Modelovanje

- Jezici za modelovanje problema planiranja: *STRIPS* i *PDDL*
 - jednostavno izražavanje problema klasičnog planiranja
 - PDDL 2.1 podržava i temporalno planiranje
 - Slaba podrška za aritmetičke uslove
- U obliku problema zadovoljenja ograničenja (SAT/SMT/MiniZinc)
 - zahteva više truda prilikom kodiranja
 - unapred fiksirana dužina plana
 - SMT i MiniZinc nude bogatiji jezik za izražavanje uslova (npr. aritmetika)

Rešavanje problema klasičnog planiranja

Rešavanje

- Specifični alati zasnovani na algoritmima informisane pretrage (poput A^*)
 - očekuju ulaz na jeziku PDDL
 - **fast-downward**
- Upotreba SAT/SMT/CSP rešavača
 - „ručno” kodiranje problema na odgovarajućem jeziku (DIMACS/SMT-LIB/MiniZinc)
 - postoje i prevodioci sa PDDL na DIMACS ili SMT-LIB
 - MiniZinc ?
 - različite vrste kodiranja (sekvencijalni, paralelni planovi i sl.).

Overview

- 1 Uvod
- 2 Klasično planiranje
- 3 Rešavanje problema klasičnog planiranja
- 4 Primeri i primene**

Primeri

Primeri

U nastavku izlaganja slede primeri nekih primena.