



10.1. DEFINICIJA KLASIČNOG PLANIRANJA

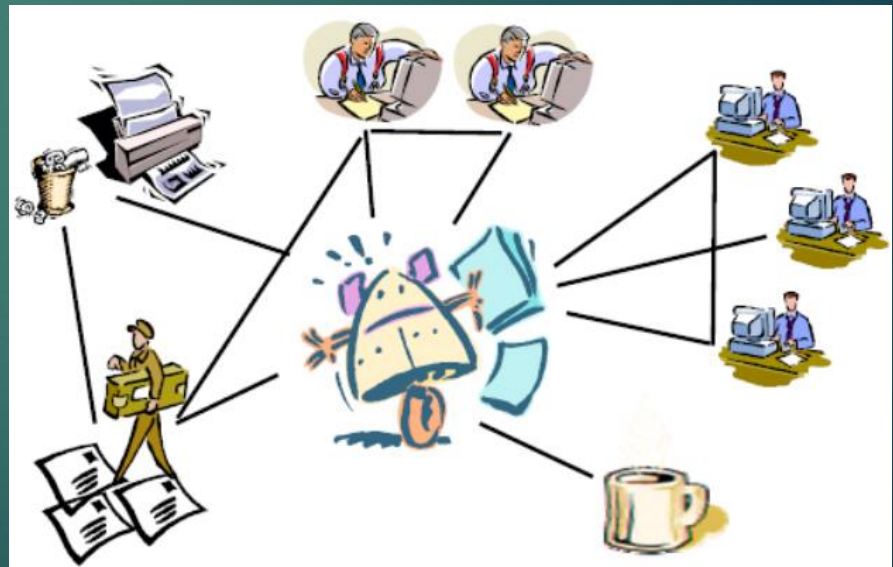
GORAN GOSARIĆ
ZAGREB, 19. SIJEČNJA 2015.

Pregled

- ▶ Uvod
- ▶ Definicija klasičnog planiranja
- ▶ Primjeri problema
- ▶ Teorijska složenost

Uvod

- ▶ Planiranje je važan dio umjetne inteligencije.
- ▶ Cilj je pokazati kako agent može iskoristiti prednost strukture problema za pronalazak plana akcija.
- ▶ U sklopu klasičnog planiranja razvija se izražajan, ali pažljivo ograničen jezik.



Definicija klasičnog planiranja

- ▶ Planiranje je proces odabira niza akcija kako bi se postigao određeni cilj.
- ▶ U klasičnog planiranju, pretpostavljamo da je okolina deterministička, statična, konačna, diskretna te uočljiva.
- ▶ Problem za Wumpusov svijet – za svaku akciju kretanja unaprijed postoje: 4 smjera, T vremena, te n^2 lokacija.
- ▶ Koristi se PDDL (*Planning Domain Definition Language*)
- ▶ Pomoću PDDL-a opisujemo: početno stanje, akcije koje su dopuštene u trenutnom stanju, rezultat primjenjenih akcija te ciljno stanje.

Stanja

- ▶ Stanje – reprezentiramo konjunkcijom pozitivnih literala, odnosno nefunkcijskih atoma logike prvog reda, kojima manipuliramo logičkim zaključivanjem, ili skup njih kojima manipuliramo skupovnim operacijama.
- ▶ Stanja ili logički uvjeti koji nisu spomenuti, smatraju se lažnima
- ▶ Primjer: $At(Truck_1, Zagreb) \wedge At(Truck_2, Split)$
- ▶ Nije dozvoljeno:
 - ▶ $At(x, y)$ – nije temeljni literali
 - ▶ $At(Father(Marko), Zagreb)$ – funkcijski simbol

Akcije

- ▶ Akcije su opisane skupom akcijskih shema koje implicitno definiraju funkcije akcije i rezultata u danom stanju.
- ▶ *Akcija*(*Leti*(*p*, *from*, *to*),
PREDUVJET: $At(p, from) \wedge Avion(p) \wedge Luka(from) \wedge Luka(to)$
EFEKT: $\neg At(p, from) \wedge At(p, to)$).
- ▶ Shema: imena akcije, varijable korištene u shemi, preduvjet i efekt.
- ▶ Skup akcijskih shema (bez varijabli), reprezentiramo jednom akcijskom shemom.
- ▶ Varijable univerzalno kvantificirane.
- ▶ PDDL specificira rezultat akcije u onome što se promjenilo, sve ostalo što je ostalo isto se ne spominje.

- ▶ Primjer leta zrakoplova:

$Akcija(Leti(P_1, Zagreb, Split)),$

PREDUVJET: $At(P_1, Zagreb) \wedge Avion(P_1) \wedge Luka(Zagreb) \wedge Luka(Split)$

EFEKT: $\neg At(P_1, Zagreb) \wedge At(P_1, Split)$.

- ▶ Preduvjeti su konjunkcije pozitivnih literala, dok su efekti su konjunktije literala, u oba slučaja bez funkcija.
- ▶ Preduvjet definira stanje u kojem akcija može biti izvršena, dok efekt definira rezultat izvršenja akcije.
- ▶ Kažemo da je akcija primjenjiva u stanju s ako s zadovoljava preduvjete.

- 
- ▶ Rezultat izvršavanja akcije a u stanju s definirano je stanjem s' koje je reprezentirano kao skup literala formiranih od s bez negativnih literala u efektu akcije, te unije pozitivnih literala u efektu akcije.

$$REZULTAT(s, a) = (s - OBRIŠI(a)) \cup DODAJ(a)$$

- ▶ Primjer.

Akcija $Leti(P_1, Zagreb, Split)$ će odbaciti $At(P_1, Zagreb)$ i dodati $At(P_1, Split)$.

- ▶ Varijable u efektu akcije moraju se također pojavljivati i u preduvjetu akcije.
- ▶ $REZULTAT(s, a)$ ima samo temeljne atome.

- 
- ▶ Primjetimo, da PREDUVJET određuje vrijeme t dok EFEKT određuje $t + 1$.
 - ▶ Skupom akcijskih shema definirali smo domenu planiranja.
 - ▶ Specifičan problem unutar domene definiran je još početnim stanjem te ciljnim stanjem.
 - ▶ Početno stanje – konjunkcija atoma
 - ▶ Ciljno stanje – konjunkcija literala koji mogu sadržavati varijable
 - ▶ Problem je riješen kada je pronađen niz akcija koje nas dovode do ciljnog stanja.

Primjer 1. Transport paketa avionom

Početno($At(C_1, Zagreb) \wedge At(C_2, Split) \wedge At(P_1, Zagreb) \wedge At(P_2, Split) \wedge Teret(C_1) \wedge Teret(C_2) \wedge Avion(P_1) \wedge Avion(P_2) \wedge Luka(Split) \wedge Luka(Zagreb)$)).

Cilj($At(C_1, Split) \wedge At(C_2, Zagreb)$)).

Akcija(*Utovari*(c, p, a),

PREDUVJET: $At(c, a) \wedge At(p, a) \wedge Teret(c) \wedge Avion(p) \wedge Luka(a)$

EFEKT: $\neg At(c, a) \wedge In(c, p)$)).

Akcija(*Istovari*(c, p, a),

PREDUVJET: $In(c, p) \wedge At(p, a) \wedge Teret(c) \wedge Avion(p) \wedge Luka(a)$

EFEKT: $At(c, a) \wedge \neg In(c, p)$)).

Primjer 1. Transport paketa avionom

Akcija(*Leti*(*p*, *from*, *to*),

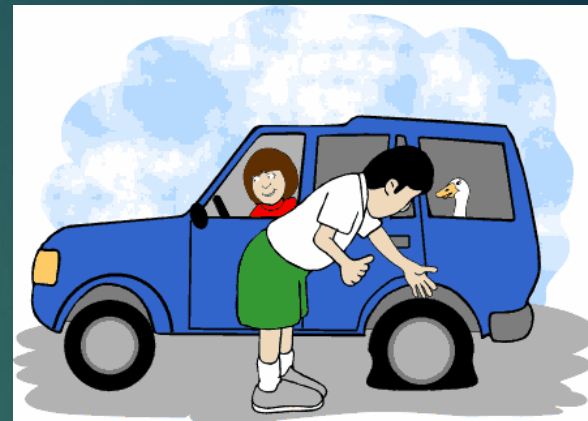
PREDUVJET: $At(p, from) \wedge Avion(p) \wedge Luka(from) \wedge Luka(to)$

EFEKT: $\neg At(p, from) \wedge At(p, to)$).

- ▶ Problem je definiran s tri akcije: Utovari, Istovari, Leti.
- ▶ $In(c, a)$ - teret c je u avionu a .
- ▶ $At(x, a)$ - avion ili teret x je u luci a .
- ▶ Greška: $Leti(P_1, Zagreb, Zagreb)$ - $from \neq to$.
- ▶ Pošto, u PDDL-u ne postoji univerzalni kvantifikator, problem da sav teret prebacimo avionom rješavamo tako da teret prestaje biti At kada je In .
- ▶ Rješenje:
 $[Utovari(C_1, P_1, Zagreb), Leti(P_1, Zagreb, Split), Istovari(C_1, P_1, Split), Utovari(C_2, P_2, Split), Leti(P_2, Split, Zagreb), Istovari(C_2, P_2, Zagreb)]$.

Primjer 2. Problem rezervne gume

- ▶ „Zamjena prazne gume automobila”.
- ▶ Ciljno stanje: dobra rezervna guma montirana na automobil.
- ▶ Početno stanje: prazna guma na automobilu i dobra rezervna u bunkeru automobila.
- ▶ Četiri akcije: Uzmi rezervnu gumu iz bunkera, skini praznu gumu s automobila, stavi rezervnu, ostavi automobil na mjestu.
- ▶ Pretpostavimo da je riječ o „opasnom” kvartu, pa bi ostavljanje automobila rezultiralo krađom.



Primjer 2. Problem rezervne gume

Početno($Guma(Prazna) \wedge Guma(Rezervna) \wedge$
 $At(Prazna, Osovina) \wedge At(Rezervna, Bunker)$).

Cilj($At(Rezervna, Osovina)$).

Akcija($Makni(objekt, lokacija)$,

PREDUVJET: $At(objekt, lokacija)$

EFEKT: $\neg At(objekt, lokacija) \wedge At(objekt, Tlo)$).

Akcija($Montiraj(t, Osovina)$,

PREDUVJET: $Guma(t) \wedge At(t, Tlo) \wedge \neg At(Prazna, Osovina)$

EFEKT: $\neg At(t, Tlo) \wedge At(t, Osovina)$).

Primjer 2. Problem rezervne gume

Akcija(OstaviAuto,

PREDUVJET:

EFEKT: $\neg \text{At}(\text{Prazna}, \text{Tlo}) \wedge \neg \text{At}(\text{Rezervna}, \text{Osovina}) \wedge$
 $\neg \text{At}(\text{Rezervna}, \text{Bunker}) \wedge \neg \text{At}(\text{Prazna}, \text{Tlo}) \wedge$
 $\neg \text{At}(\text{Prazna}, \text{Osovina}) \wedge \neg \text{At}(\text{Prazna}, \text{Bunker}).$

► Rješenje:

$[\text{Makni}(\text{Prazna}, \text{Osovina}), \text{Makni}(\text{Rezervna}, \text{Bunker}),$
 $\text{Montiraj}(\text{Rezervna}, \text{Osovina})].$

Teorijska složenost

- ▶ PlanSAT – postoji li uopće plan kojim rješavamo problem planiranja.
- ▶ Bounded PlanSAT – postoji li plan duljine k ili manje.
- ▶ Oba ova problema su odlučiva za klasično planiranje, a to proizlazi iz činjenice da je broj stanja konačan.

Sažetak

- ▶ Klasično planiranje je proces pronalaženja niza akcija kojima iz početnog stanja stižemo u ciljno stanje, uz pretpostavku da je okolina deterministička, statična, konačna, diskretna te uočljiva.
- ▶ Agent može predvidjeti posljedice akcije.
- ▶ Nema eksternih događaja koje kontroliraju stanja.
- ▶ Iz jednog stanja prelazimo u drugo stanje.

Literatura

- ▶ S. Russell, P. Norvig, *Artificial Intelligence, A Modern Approach*, Pearson Education, New Jersey, 2010.
- ▶ L. Sinapova, *CmSc310 Artificial Intelligence, Classical Planning*, dostupno na:
http://faculty.simpson.edu/lydia.sinapova/www/cmsc310/LN310_AIMA/L10-Planning.pdf (siječanj 2015.)
- ▶ *Planning, Prentice hall*, dostupno na:
<http://aima.cs.berkeley.edu/2nd-ed/> (siječanj 2015.)