

Algoritmi za planiranje kao pretraživanje prostora stanja

Domagoj Beti

PMF - Matematički odsjek

Zagreb, 19. siječnja 2015.

Sadržaj

- ▶ Pretraživanje prostora stanja unaprijed i unazad
- ▶ Heuristike za planiranje

Pretraživanje prostora stanja unaprijed

- ▶ Od oko 1961.-1998. godine pretraživanje unaprijed se smatralo nedovoljno efikasno da bi se izvodilo u praksi.
- ▶ Počinjemo od inicijalnog i krećemo se prema ciljnom stanju.
- ▶ Sklonost pretraživanja irelevantnih akcija.
 - ▶ Akcijska shema *Buy(isbn)* sa efektom *Own(isbn)* predstavlja 10 trilijuna osnovnih akcija.
 - ▶ Algoritam za neinformirano pretraživanje unaprijed bi pretraživao te akcije dok ne bi pronašao onu koja vodi do cilja.

Pretraživanje prostora stanja unaprijed (2)

- ▶ Često imaju (pre)veliki prostor stanja.
 - ▶ Za problem transporta paketa avionom uzmimo da imamo 10 aerodroma, svaki od njih ima 5 aviona i 20 paketa. Cilj je premjestiti sve pakete iz aerodroma A u aerodrom B .
 - ▶ Svaki od 50 aviona može letjeti u neki od preostalih 9 aerodroma i svaki od 200 paketa se može utovariti(istovariti) u svaki od aviona na aerodromu na kojem se nalazi.
 - ▶ U svakom stanju imamo minimum od 450 i maksimum od 10450 akcija.
 - ▶ Uzmemo li da je prosječno 2000 mogućih akcija u svakom stanju, graf pretraživanja u dubini rješenja ima 2000^{41} čvorova.

Pretraživanje prostora stanja unatrag

- ▶ Počinjemo od ciljnog i krećemo se unatrag dok ne nađemo redoslijed koraka s kojima dolazimo do početnog stanja.
- ▶ Nazivamo ga pretraživanjem relevantnih stanja jer u obzir uzimamo samo one akcije koje su relevantne za cilj (ili trenutno stanje).
- ▶ Neka je g osnovni opis cilja i a osnovna akcija, regresija iz g preko a jest opis stanja g' koju definiramo kao
$$g' = (g - ADD(a)) \cup Precond(a).$$

Pretraživanje prostora stanja unatrag (2)

- Trebamo imati i parcijalno instancirane akcije i stanja (ne samo osnovne).

Primjer

Prepostavimo da je cilj dostaviti određeni paket u Split:

$At(C_2, Split)$. To nam ukazuje na akciju $Unload(C_2, p', Split)$:

$Action(Unload(C_2, p', Split),$

PRECOND:

$In(C_2, p') \wedge At(p', Split) \wedge Cargo(C_2) \wedge Plane(p') \wedge Airport(Split)$

EFFECT:

$At(C_2, Split) \wedge \neg In(C_2, p')$.

$$g' = In(C_2, p') \wedge At(p', Split) \wedge Cargo(C_2) \wedge Plane(p') \wedge Airport(Split)$$

Pretraživanje prostora stanja unatrag (3)

- ▶ Kako odrediti akcije koje su kandidati za regresiju?
- ▶ Hoćemo akcije koje su relevantne za cilj - barem jedan efekt akcije se mora unificirati sa elementom cilja.
- ▶ Akcija ne smije imati nijedan efekt koji negira element cilja.
 - ▶ Ako je cilj $A \wedge B \wedge C$ a akcija ima efekt $A \wedge B \wedge \neg C$ ona nije relevantna jer nam treba bar još jedan korak da postignemo C

Pretraživanje prostora stanja unatrag (4)

- ▶ Pogledajmo cilj $Own(0136042597)$ i akcijsku shemu $A = Action(Buy(i), PRECOND: ISBN(i), EFFECT: Own(i))$.
- ▶ Unificiramo cilj $Own(0136042597)$ sa efektom $Own(i')$ i dobijemo supstituciju $\theta = \{i'/0136042597\}$ te napravimo regresiju preko akcije $Subst(\theta, A')$ da bi dobili opis prethodnog stanja $ISBN(0136042597)$ što je dio početnog stanja pa smo gotovi.
- ▶ Zbog toga što gledamo skupove stanja teško je osmisлити dobru heuristiku - zbog toga ipak prednost ima pretraživanje unaprijed.

Heuristike za planiranje

- ▶ Pretraživanja unaprijed i unatrag su efikasna samo uz dobre heuristike.
- ▶ Promatramo li problem pretraživanja kao graf, pri čemu su vrhovi stanja, a bridovi akcije, problem svodimo na pronalaženje puta između početnog i ciljnog stanja.
- ▶ Problem možemo "opustiti" i učiniti ga jednostavnijim:
 - ▶ dodavanjem vrhova u graf
 - ▶ grupiranjem vrhova tako da dobijemo prostor sa manje stanja

Heuristike za planiranje - dodavanje vrhova u graf

- ▶ Heuristika koja ignorira listu brisanja.
 - ▶ Iz svih akcija maknemo listu brisanja (tj. sve negativne literale iz efekta).
 - ▶ Monotono kretanje prema cilju - nijedna akcija neće prebrisati napredak druge akcije.
 - ▶ Pronalazak optimalnog rješenja ovog "opuštenog" problema je NP-težak, ali se približno rješenje može pronaći hill-climbingom.

Heuristike za planiranje - dodavanje vrhova u graf

- ▶ Heuristika koja ignorira preduvjete.
 - ▶ Svaka akcija se može dogoditi u svakom stanju - svaki fluent cilja se može dobiti u jednom koraku.
 - ▶ Heuristike uzimaju u obzir da neke akcije mogu zadovoljiti više ciljeva a ignoriraju da neke akcije mogu prebrisati efekte drugih.
 - ▶ "Opustimo" problem ignorirajući sve preduvjete i efekte osim onih čiji su literali u cilju te izračunamo minimalni broj potrebnih akcija td. njihova unija zadovoljava cilj.
 - ▶ NP-težak problem, njegovo rješenje ne garantira dopustivost heuristike.

Heuristike za planiranje - apstrakcija stanja

- ▶ Ignoriranje nekih fluenata.
 - ▶ U primjeru transporta paketa avionom smanjimo broj stanja sa 10^{155} na 10^{17} .
 - ▶ Rješenje će biti kraće nego u originalnom prostoru - imamo dopustivu heuristiku.
- ▶ Ključna ideja u definiranju heuristika je dekompozicija.
- ▶ Pretpostavka nezavisnosti podciljeva - trošak rješavanja konjunkcije podciljeva je aproksimiran sumom troška rješavanja svakog od njih nezavisno.
 - ▶ Može biti optimističan i pesimističan.

Heuristike za planiranje - apstrakcija stanja (2)

Primjer

Pretpostavimo da je cilj skup fluenata $G = G_1 \cap \dots \cap G_n$ te nađemo planove $P_1, \dots, P_n$ koji rješavaju podciljeve $G_1, \dots, G_n$.

Koja je procjena troška da bi postigli G ?

Ako promatramo $Cost(P_i)$ kao heurističku procjenu znamo da je $\max_i Cost(P_i)$ jedna dopustiva heuristika.

Ako možemo zaključiti da su G_i i G_j nezavisni, tj. efekti od P_i ne mijenjaju preduvjete i ciljeve od P_j tada je heuristika $Cost(P_i) + Cost(P_j)$ dopustiva i točnija od maksimuma.

Heuristike za planiranje - apstrakcija stanja (3)

- ▶ Jasno je da apstrakcija stanja ima veliki potencijal u smanjivanju prostora stanja, trik je "samo" pronaći onu kojoj je ukupni trošak - definiranja apstrakcije, pretraživanja i preslikavanja nazad u originalni problem manji od troška rješavanja početnog problema.
- ▶ FF (FastForward).

Literatura

- ▶ S.Russell, P.Norvig, *Artificial Intelligence, A Modern Approach*, Pearson Education, New Jersey, 2010.