

Ekspertski sistemi

Lekcija 2: Pretraživanje

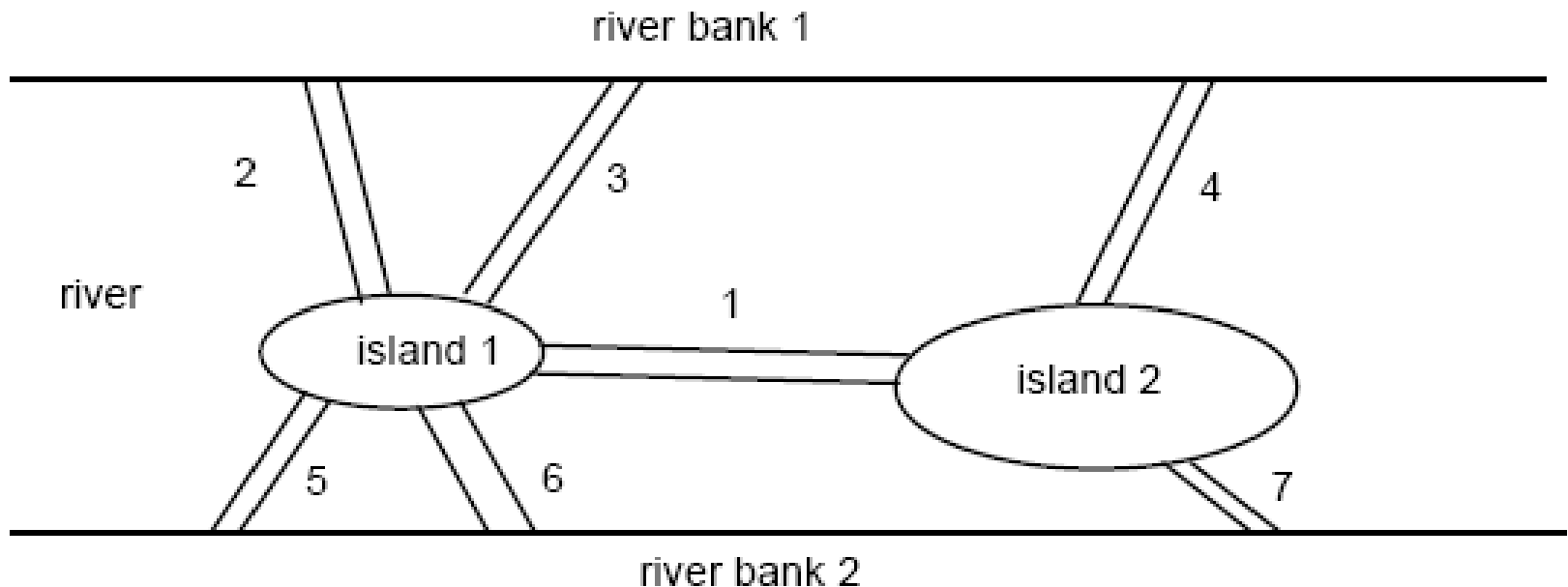
Pretraživanje kao metoda rešavanja problema

- Pretraživanja se odnosi na nalaženje nekog podatka u skupu podataka organizovanih na određeni način, a na osnovu određene karakteristike tog podatka.
- Takvo pretraživanje je određeno sledećim parametrima:
 - organizacijom i vrstom pristupa podacima,
 - karakteristikom podatka koji se traži,
 - strategijom pretraživanja

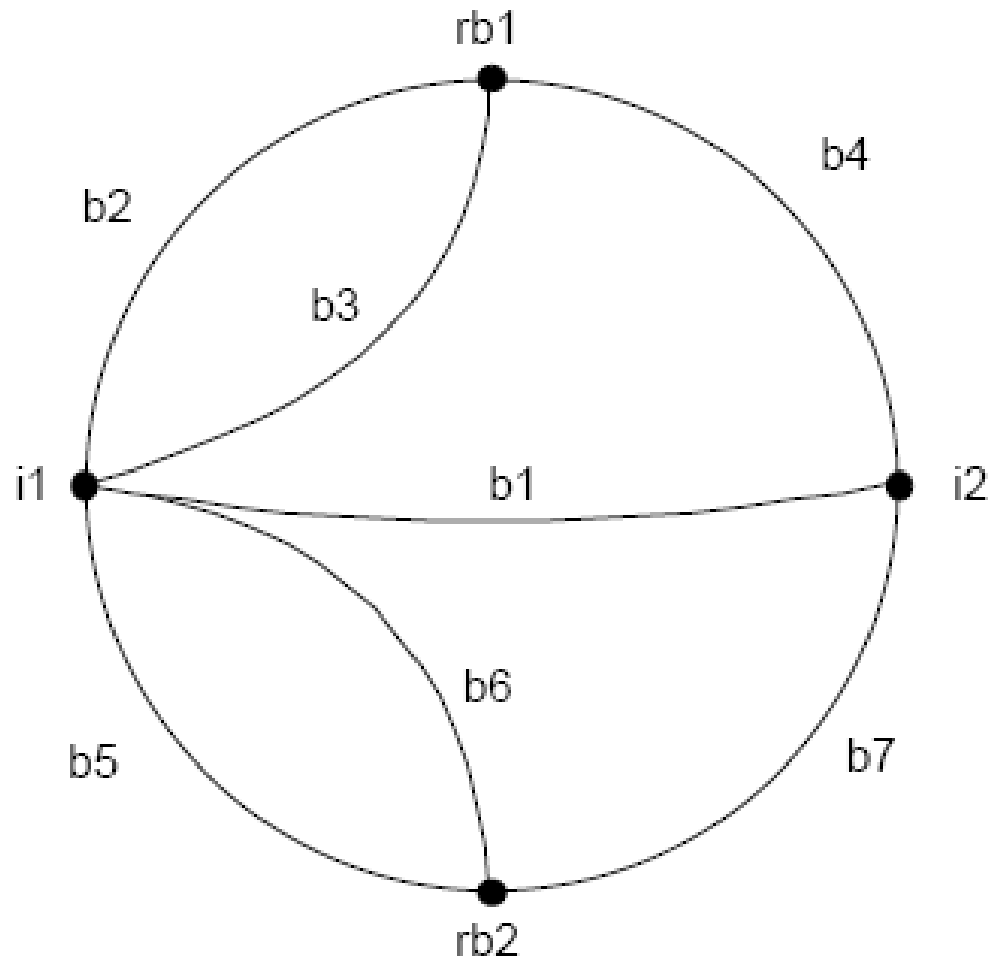
Pretraživanje kao metoda rešavanja problema

- Pretraživanje u ekspertskim sistemima ima drugačiji smisao. Ono se sastoji u nalaženju rešenja nekog problema do koga se inače teško može doći na opšti način primenom metoda klasičnog programiranja.
- Primer: putovanje iz jednog mesta u drugo. Od jednog mesta do drugog može postojati više puteva različitih dužina i oni se mogu prikazati u vidu stabla
- Ako bi se računaru dostavio opis ove putne mreže, njegov zadatak bi bio da izabere onaj put od jednog mesta do drugog koji je najkraći.
- Problem bi se mogao postaviti u složenijoj formi, ako bi uključili kriterijum izbora kvaliteta puta i njegove opterećenosti

Pretraživanje kao metoda rešavanja problema



Pretraživanje kao metoda rešavanja problema



Pretraživanje kao metoda rešavanja problema

- Može i u vidu mreže, koja se dalje može prevesti u oblik stabla. Ova transformacija mora biti ekvivalentna sa tačke gledišta pretraživanja, to znači, da važi

za svako $C \{r_i \mid i = 1, 2, \dots\}$

gde je C cilj, a $\{r_i \mid i = 1, 2, \dots\}$ skup rešenja

- Model pretraživanja zasnovan na strukturi stabla je jednostavniji od mrežnog modela, međutim transformacija mrežnog modela u model stabla nije uvek jednostavna.

Pretraživanje kao metoda rešavanja problema

- Primer 2: dve posude od 4 i 3 litra bez mernih oznaka. Kako se posuda od 4 litra može napuniti tačno do polovine svoje zapremine?
- Primer 3: igranje šaha. Cilj bi se mogao definisati kao ostvarivanje pozicije na šahovskoj tabli u kome protivnik nema na raspolaganju dozvoljen potez i njegov kralj je napadnut.

Pretraživanje kao metoda rešavanja problema

- definisati precizno problem, odrediti njegove početne pozicije koje čine prihvatljivo rešenje problema,
- analizirati problem u cilju otkrivanja njegovih bitnih svojstava koje mogu da budu odlučujuće u izboru metode za njegovo rešavanje,
- odabrati najbolju metodu i primeniti je u rešavanju problema

Definisanje problema pretraživanja

- stanjima,
- operatorima,
- strategijama pretraživanja,
- procenjivačkim funkcijama,
- faktorom grananja

Stanja i operatori

- Stanje se može definisati kao snimak procesa u vremenu
- Od značaja su stabilna stanja, to jest stanja, koja bez spoljnih stimulansa ostaju nepromenjena.
- Promene koje se dešavaju pod dejstvom spoljnih stimulansa su trenutne i diskretne.
- Da bi stanja bila od koristi u procesu pretraživanja, ona moraju da u sebe uključe sve što je važno o nekoj situaciji koja se analizira, što obično znači listu činjenica.
- Promena stanja pod dejstvom spoljnog stimulansa zove se prelaz.
- Cilj pretraživanja se stoga svodi na to da se polazeći od početnog, startnog stanja, nizom prelaza ono prevede u završno stanje koje predstavlja cilj, odnosno rešenje problema.

Stanja i operatori

- Struktura prostora stanja odgovara strukturi rešavanja problema iz dva razloga:
 - omogućava formalnu definiciju problema kao potrebu da se neka data situacija prevede u željenu situaciju primenom dozvoljenih operacija,
 - dozvoljava da se rešavanje problema svede na kombinovanje i primenu određenih operatora u prostoru stanja i pretraživanje, kao opšte tehnike ispitivanja stanja sa ciljem da se nađe putanja od tekućeg do ciljnog stanja

Direktne tehnike i veštačka inteligencija

- **Prednosti**

- zbog ugrađenog procesa rezonovanja, mogu se rešavati vrlo složeni problemi
- sposobnost adaptacije pri izmeni u specifikaciji problema; zbog elastičnosti dinamičkog procesa rezonovanja, programi koji imaju u sebi ugrađen proces rezonovanja mogu da obave deo svog zadatka i u uslovima nepotpunih ili netačnih podataka
- razdvajanjem znanja od procesa rezonovanja, obezbeđuje se mogućnost izmene znanja, a da se pritom ne dira u proces rezonovanja,
- ugrađivanjem procesa rezonovanja u programe on postaje vidljiv za korisnika, što poboljšava razumevanje programa

Direktne tehnike i veštačka inteligencija

- **Nedostaci:**

- Iako je proces rezonovanja ugrađen u program, teško je dati unapred opis programskih akcija, jer se one određuju dinamički
- Mada se u mnogim slučajevima mogu pronaći načini za rešavanje problema, dešava se da je teško opisati način na koji se do rešenja došlo
- Realizacija rešavanja problema je manje efikasna u izvršenju nego kod direktnih metoda zbog značajnog režijskog vremena koje se zahteva pri pretraživanju
- Za mnoge stvarne probleme potreban memorijski prostor je suviše veliki da bi se moglo ostvariti iscrpno pretraživanje, praktično gornja granica za iscrpno pretraživanje je oko $10!$ mogućih rešenja

Operatori

- Neka je data tačka X u prostoru stanja $PS(sx_1, sx_2, sx_3, \dots, sx_n)$, gde su $sx_1, sx_2, sx_3, \dots, sx_n$ deskriptori tačke X u prostoru stanja.
- Tačka X se može prevesti u tačku Y opisanu sa $PS(sy_1, sy_2, sy_3, \dots, sy_n)$ pod dejstvom nekog operatora op_i iz skupa operatora

$$\underline{op}_i \quad X \rightarrow Y$$

- Operator op_i ne može se uvek bezuslovno primeniti na svako X , znači ne važi

$$\underline{op}_i \quad X \rightarrow Y \text{ za svako } X$$

- op_i se može primeniti na tačku X u prostoru stanja samo ako deskriptori tačke X zadovoljavaju podskup uslova iz skupa uslova U

Formalni opis problema

- Definirati prostor problema koji će obuhvatiti sve njegove moguće tačke. Definiranje PS treba učiniti ako je to moguće bez nabrojanja svih stanja koje ono sadrži
- Definirati početna stanja tačke od kojih započinje rešavanje problema. Ovih stanja može da bude jedno ili više
- Definirati jedno ili više završnih stanja koje predstavljaju prihvatljivo rešenje problema, ta stanja obeležiti kao ciljna stanja (tačke)

Primer – posude sa vodom

- Date su dve posude od 4 i 3 litra

Water jug operations

1. $(x, y \mid x < 4)$	$(4, y)$	<u>Fill 4</u>	α move
2. $(x, y \mid y < 3)$	$(x, 3)$	<u>Fill 3</u>	
3. $(x, y \mid x > 0)$	$(0, y)$	<u>Dump 4</u>	β move
4. $(x, y \mid y > 0)$	$(x, 0)$	<u>Dump 3</u>	
5. $(x, y \mid x+y \geq 4 \text{ and } y > 0)$	$(4, y - (4 - x))$	<u>Pour from 3 to 4 until 4 is full</u>	
6. $(x, y \mid x+y \geq 3 \text{ and } x > 0)$	$(x - (3 - y), 3)$	<u>Pour from 4 to 3 until 3 is full</u>	
7. $(x, y \mid x+y \leq 4 \text{ and } y > 0)$	$(x+y, 0)$	<u>Pour all water from 3 to 4</u>	

Primer – posude sa vodom

- Jedno rešenje

Water Jug Problem: one solution

<u>Gallons in y</u>	<u>Transition Rule</u>
0 2	fill 3
3 7	pour from 3 to 4
0 2	fill 3
3 5	pour from 3 to 4 until 4 is full
2 3	dump 4
2 7	pour from 3 to 4
0	

Primer - novčići

- Data su dva para novčića: srebrni i zlatni, Novčići su poređani jedan pored drugog sa razlikom između: SSPZZ. Uspostaviti njihov položaj ZZPSS.

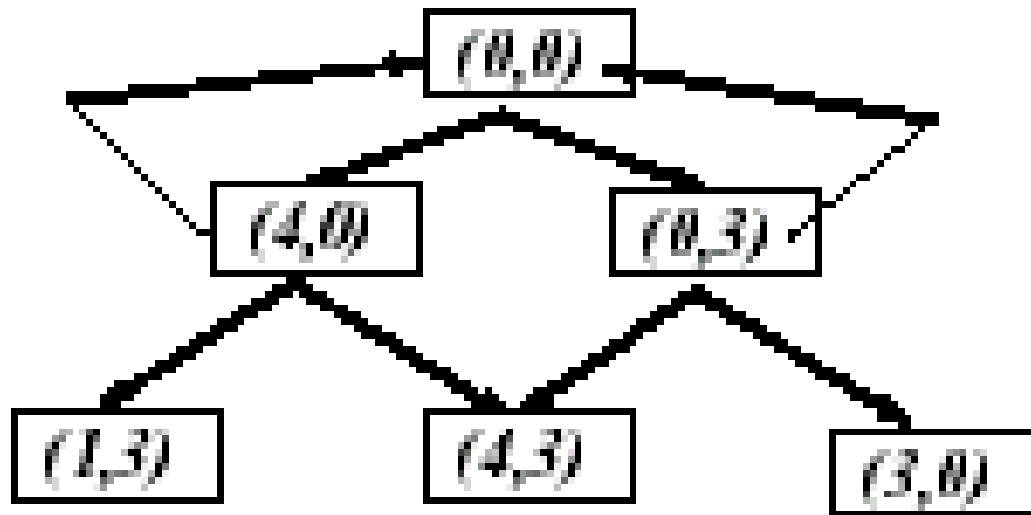
1	pomeranje udesno	samo S
2	pomeranje ulevo	samo Z
3	preskok udesno za jedno mesto	samo S
4	preskok ulevo za jedno mesto	samo Z

Graf kao struktura pretraživanja

- uzeti početno stanje kao startni čvor grafa,
- primenom nekog od raspoloživih operatora generisati novu tačku u prostoru stanja koja predstavlja čvor sledbenik tačke iz koje je generisana,
- čvor roditelj se spaja granom sa čvorom sledbenikom, pri čemu se grana označava imenom operatora koji ga je formirao,
- definisati ciljno stanje
- proces generisanja čvorova se nastavlja od poslednje generisane tačke u PS, sve dok se ne dođe do ciljnog stanja, čime se dobija potpuna putanja od startnog do ciljnog stanja politika generisanja čvorova čini strategiju ograničavanja ili kontrolnu strategiju.

Graf - posude

Water jug problem: graph



Strategije pretraživanja

- Kako se odvija proces pretraživanja?
- Kojim redosledom se vrši razvijanje čvorova?
- Strategije pretraživanja treba da zadovolje:
 - da obezbeđuju kretanje u grafu pretraživanja
 - da budu sistematične

Strategije pretraživanja

- Dat je skup od N gradova, koje trgovački putnik mora da poseti samo jedanput. Između svakog para gradova postoji direktna putna veza. Potrebno je pronaći najkraću putanju koju trgovački putnik treba da sledi, da bi se, polazeći iz bilo kog grada, ponovo vratio u njega
- Broj gradova = N , Broj različitih putanja između gradova = $(N-1)!$. Ispitivanje jedne putanje je proporcionalno sa N . Ukupno vreme reda $O(N!)$. Za $N=10$, dobija se broj 3268800.
- Sa vremenske tačke gledišta, problem postaje nerešiv, jer dovodi do kombinatorne eksplozije.

Opšti pristupi i elementi u pretraživanju

- direktno ulančavanje,
- povratno ulančavanje,
- hibridno ulančavanje,
- uređivanje komponenti ulančavanja,
- konkurentna obrada pravila,
- izbor ulančavanja,
- topologija pretraživanja,
- heurističko pretraživanje,
- funkcija procene,
- funkcija cene koštanja,
- faktor grananja,
- uparivanje i rezolucija ciljeva,
- verovatnoća pretraživanja

Direktno ulančavanje

- Polazeći od startne tačke u prostoru stanja, naći putanju do ciljne tačke.
- Direktno ulančavanje se može koristiti i za izvođenje određenih zaključaka na osnovu datih činjenica.
- Skup činjenica bi predstavljao startno stanje, a zaključak bi predstavljao ciljno stanje.
- Ciljna tačka ne mora unapred da bude poznata, već se do nje dolazi samim direktnim ulančavanjem

Primer - zebra

Postoji 5 kuća, od kojih je svaka obojena različitom bojom i u koj stanuju ljudi različitih nacionalnosti, sa različitim životinjama, pićem i cigaretama. Data su sledeća ograničenja:

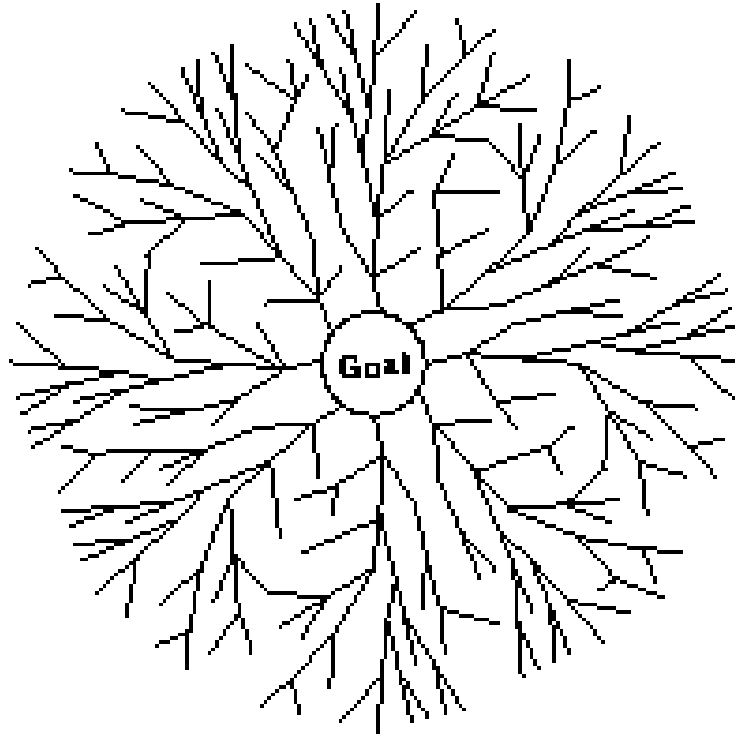
1. Englez živi u crvenoj kući.
2. Jugosloven ima psa.
3. Kafa se pije u zelenoj kući.
4. Šumadinac pije čaj.
5. Zelena kuća je neposredno desno od bele kuće.
6. Pušač Drave gaji krave.
7. Cigarete Morava se puše u žutoj kući.
8. Mleko se pije u središnjoj kući.
9. Norvežanin živi u prvoj kući gledajući sa leve strane.
10. Čovek koji puši Chesterfild živi u kući koja je susedna kući u kojoj živi čovek sa lisicom.
11. Cigarete Morava se puše u kući koja je susedna kuću u kojoj čovek drži konje.
12. Pušač Zete pije đus.
13. Japanac puši Lord.
14. Norvežanin živi u kući koja je susedna plavoj kući.

Treba odgovoriti na sledeća pitanja: Ko ima zebra? Ko pije vodu?

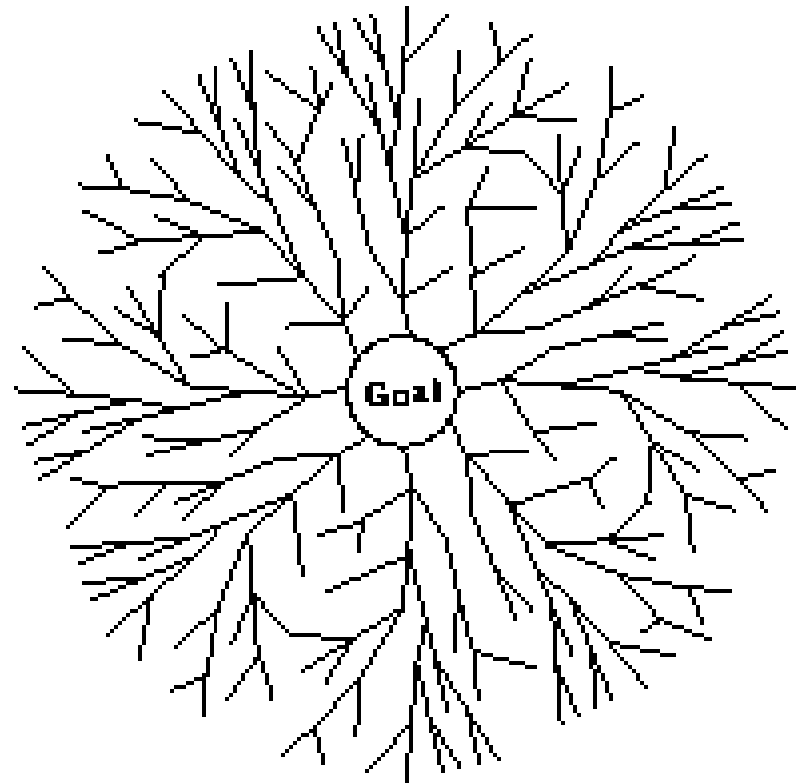
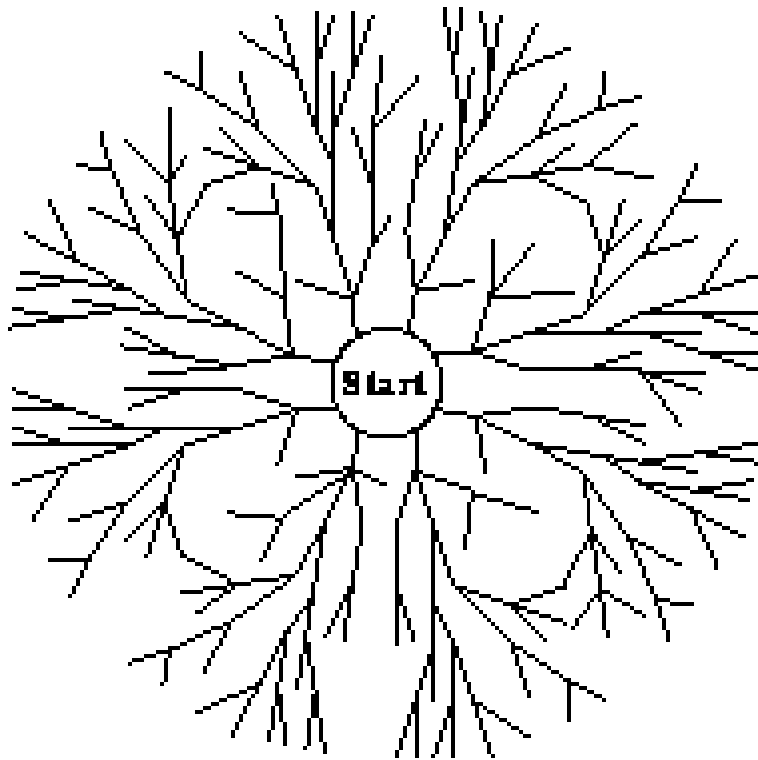
Povratno ulančavanje

- Polazeći od ciljne tačke u prostoru stanja, naći putanju do startne tačke
- Efikasnost povratnog ulančavanja se može poboljšati na dva načina:
ubacivanjem izvedenih činjenica u već dobijenu bazu činjenica i izdvajanjem virtuelnih činjenica.

Povratno ulančavanje



Hibridno ulančavanje



(from AIM A Figure 3.17)

Izbor ulančavanja

- Da li ima više startnih ili ciljnih stanja? Poželjnije je krenuti od manjeg ka većem broju stanja.
- U kom smeru je prosečan broj čvorova kroz koje se mora proći manji?
- Da li će se od programa tražiti da objasni svoj proces zaključivanja? Ako će se to zahtevati, bira se onaj smer koji odgovara načinu na koji korisnik razmišlja.

Izbor ulančavanja

- Kako pronaći put od kuće do nekog nedovoljno poznatog mesta u gradu i u kom smeru je to lakše odrediti? Povratno ulančavanje
- Aritmetički izrazi sa zagradama – Direktno ulančavanje
- Dokazivanje teorema u nekoj oblasti matematike - Povratno ulančavanje

Uređivanje komponenti ulančavanje

Svrha uređivanja komponenti ulančavanja je smanjivanje prostora pretraživanja, odnosno dolaženje do ciljeva (činjenica), što je posebno značajno u sistemima realnog vremena. Postoje dva osnovna principa uređivanja:

- po prioritetu,
- po specifičnosti

Uređivanje komponenti ulančavanje

Po specifičnosti: da li uslovi na desnoj strani nekog pravila i čine podskup uslova na desnoj strani pravila j.

Zahteva da na prvo mesto dođe pravilo sa najužom specifikacijom.

- /*R1*/ x: -a
- /*R2*/ y: -b
- /*R3*/ z: -a,b,c
- /*R4*/ r: -c,d
- /*R5*/ s: -a
- /*R6*/ t: -a,c

Pravila se mogu urediti po specifičnosti, tako da se dobije uređenje R3, R6, R4, R1, R2, R5.

Uređivanje komponenti ulančavanje

Problemi:

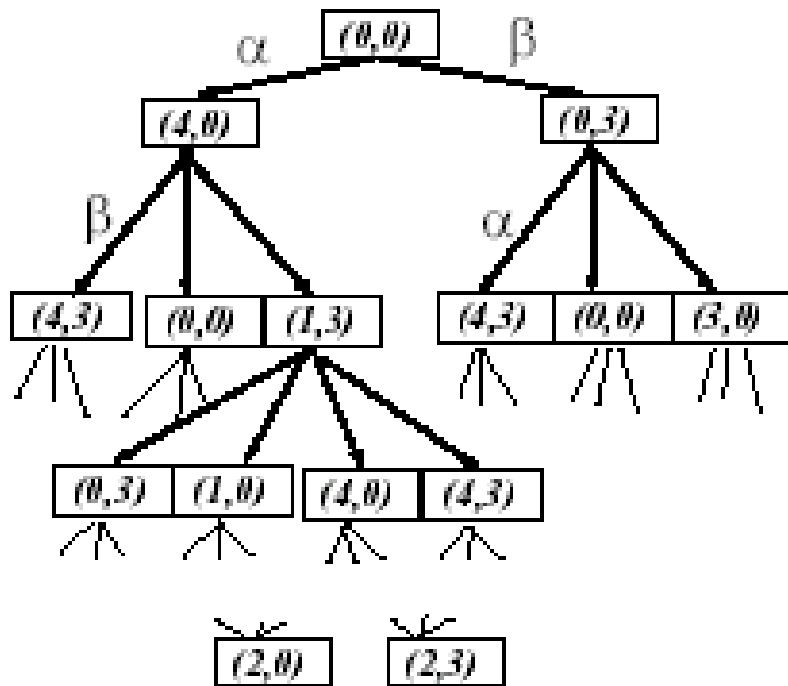
- mali je broj slučajeva u kojima postoji specifična međuzavisnost između pravila,
- pravila sa vrlo uskom specifikacijom se retko primenjuju, te njihovo stavljanje na čelo liste znači gubitak vremena.
- Drugi problem se rešava tako što se pravila širom specifikacijom proširuju sa not izrazima čime se one uže specificiraju, te pomeraju na čelo liste.
- Na primer pravilo R1 x: -a, not(b), not(c).
- Ako upiti sadrže i operatore izrazi u upitu, kod povratnog ulančavanja, čiji je pozitivan ishod najmanje verovatan treba staviti na čelo liste.

Konkurentna obrada pravila

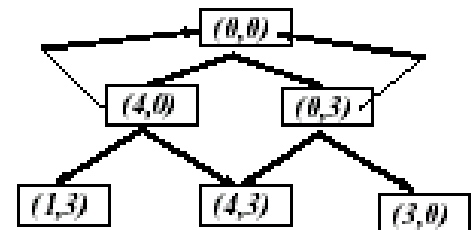
- particijski paralelizam (formiranje particija pravila, ulančavanja se vrše zasebno, rezultati se međusobno razmenjuju)
- I paralelizam (za predikatske izraze na desnoj strani pravila , koji ne vezuju promenljive ili ako ih vezuju, to čine sa različitim promenljivama)
- ILI paralelizam (pravila čije leve strane imaju isto ime predikata)
- paralelizam uparivanja promenljivih (prilikom međusobnog uparivanja dva predikatska izraza)

Topologija pretraživanja

Water jug problem: tree



Water jug problem: graph



Heurističko pretraživanje

- Osnovni problem u pretraživanju je smanjivanje prostora pretraživanja.
- Zato se mora u pretraživanje ugraditi određeno znanje, iskustvo, koje će pomoći da se taj prostor smanji.
- Heuristika u pretraživanju obezbeđuje:
 - eliminisanje celih grana iz procesa pretraživanja
 - izbor opšte putanje koju treba slediti
 - izbor sledećeg čvora za razvijanje

Heurističko pretraživanje

- Heuristika je lokalnog značaja, jer ne daje neku opšti opštu strategiju pretraživanja.
- U nekim slučajevima, može da dovede do izbora pogrešne putanje koja vodi ka rešenju.
- Osim toga, može se desiti da zbog primene heuristike odlična putanja promakne.
- Ali, i pored svega toga, heuristika poboljšava u proseku kvalitet pretraživanja smanjujući prostor pretraživanja.

Heurističko pretraživanje

- Opštenamenska heuristika se može dopuniti specifičnom heuristikom, koja proizilazi iz posebnog poznavanja određenog problema.
- Algoritam **najbližeg susedstva** - izbor sledećeg čvora u prostoru stanja, koji je, u lokalnom pogledu, bolji.
- Problem putujućeg trgovca:
 - izabрати proizvoljan grad kao startnu tačku;
 - repeat
 - izabрати sledeći grad koji je najbliži trenutnom, a koji još nije posećen;
 - until (svi gradovi posećeni)
- Ova procedura se izvršava u vremenu proporcionalnom sa N^2

Heurističko pretraživanje

- Kontrola gustine eksplozije čvorova:

$g(\text{sledbenik}) = g(\text{prethodnik}) - g(\text{rang nekog prethodnika između više prethodnika})$

- gde je:

$g(\text{sledbenik})$ - broj sledbenika koje treba zadržati u izabranom čvoru prethodnika (pri ekspanziji nekog čvora);

$g(\text{prethodnik})$ - broj čvorova prethodnika koji su zadržani u daljem razmatranju;

$g(\text{rang nekog prethodnika između više prethodnika})$ - rang verodostojnosti prethodnika između više prethodnika nekog čvora.

Heuristička funkcija procene

- Heuristička funkcija procenjuje pojedinačna stanja problema i određuje koliko su ona povoljna.
- Izbor elementa stanja koja se razmatraju, način njihove procene i težine koje dodeljuju pojedinim aspektima stanje vrše se i na takav način da heuristička funkcija u datom čvoru procesa pretraživanja da njegovu procenu iz koje se može zaključiti koliko je on dobar i da li se nalazi na traženoj putanji ka rešenju.
- Dobro odmerena heuristička funkcija može da posluži kao dobar vodič u procesu pretraživanja čineći ga efikasnijim.
- Vrednost ove funkcije nije u svim problemima na isti način povezana sa kvalitetom putanje.

Heuristička funkcija procene

- Suština funkcije procene je da upravlja procesom pretraživanja u najpovoljnijem smeru ka dostizanju cilja.
- Što heuristička funkcija daje bolju procenu kvaliteta čvora u traženju rešenja, efikasniji i uspešniji će biti put ka takvom rešenju.
- Cena izračunavanja funkcije procene može da bude veća od uštede u vremenu pretraživanja.
- Mora se voditi računa o kompromisu između dobitka i gubitka koji prozilazi iz jedne takve primene.

Funkcija cene koštanja

- Funkcija cene koštanja je retrospektivna.
- Ona daje cenu koštanja po kojoj se prelazi iz jednog stanja problema u drugo pod dejstvom nekog operatora. Ukupna cena koštanja:

$C_{tot} = \text{Suma od } c_{ij} \text{ i, j pripadaju putanji do } k$

gde je

c_{ij} - cena koštanja prelaska iz stanja problema i u stanje problema j

k - krajnja tačka putanje do koje se posmatra ukupna cena koštanja

Funkcija cene koštanja

- U pretraživanje se mogu uključiti obe funkcije tako što će se njihove vrednosti sabrati i dobiti poboljšanja u procesu pretraživanja

$$S = H + C_{\text{tot}}$$

- kod direktnog ulančavanja, računarsko vreme potrebno da se obradi neki skup činjenica
- kod planiranja puta kroz grad, C_{tot} bi bila ukupna dužina pređenog puta do posmatrane tačke
- kod igranja šaha, vreme između dva poteza
- kod zagonetke sa poljima brojeva, C_{tot} bi bio ukupan broj pomeranja brojeva do posmatrane konfiguracije brojeva.

Faktor grananja

- Kao neka mera težine problema pretraživanja izražena je kroz broj stanja kroz koje prolazi da bi se došlo do cilja. Za ovu svrhu postoje dve metode:
 - procena veličine prostora pretraživanja
 - izračunavanje srednje vrednosti faktora grananja
- Po prvoj metodi se kao gornja granica broja stanja koja se moraju ispitati da bi se rešio problem uzima broj mogućih stanja u celokupnom problemu. Skup svih mogućih stanja se naziva prostor pretraživanja.

Faktor grananja

- kod popravke kola moguća su sledeća stanja svakog dela
 - deo je u kolima i ispravan
 - deo je van kola i ispravan
 - deo je u kolima i neispravan
 - deo je van kola i neispravan
- Ako u kolima ima 2000 delova, veličina prostora pretraživanja je 4^{2000} .
- Mora se pozvati u pomoć heuristika ili heuristička funkcije procene

Faktor grananja

- Faktor grananja je dat brojem sledbenika koji se mogu generisati iz nekog čvora.
- Ako se faktor grananja ne razlikuje mnogo od stanja do stanja, onda se može govoriti o srednjoj vrednosti faktora grananja.
- Moguće je proceniti broj stanja na nivou K u prostoru grafa kao B^K , gde je B srednja vrednost faktora grananja.
- Zbog kombinatorne eksplozije ako je moguće proceniti nivo na kome se nalazi cilj moguće je proceniti broj stanja kroz koja će se proći sabirajući procenu broja stanja na svakom nivou.

Faktor grananja

- Problem prolaska kroz grad iz jedne u drugu njegovu tačku.
- Faktor grananja je oko 3 (levo, desno, pravo) za svaku raskrsnicu (čvor)
- Ukupan broj stanja za 10 raskrsnica je

$$N_{\text{tot}} = \sum 3^i = (3^{11} - 1)/(3 - 1) = 88573, \text{ gde je } i \text{ između } 0 \text{ i } 10$$

$$N_{\text{tot}} = (B^{K+1} - 1)/(B - 1)$$

Faktor grananja

- Dobitak (G) ostvaren uvođenjem u proces pretraživanja heuristike, se može izraziti sa:

$$G = Bb / Bs$$

Bb - faktor grananja bez heuristike i heurističke funkcije procene,

Bs - faktor grananja sa heuristikom i heurističkom funkcijom procene