Uvod

1. Što su ekspertni sustavi (ES) i što im je temeljni sadržaj ?
To su programski produkti namijenjeni rješavanju složenih problema u uskoj domeni primjene.
Temeljni sadržaj ekspertnih sustava je predstavljanje i obradba znanja.
2. Koja četiri pristupa oblikovanju ES razlikujemo ?
-Sustav koji rasuđuje poput ljudi(modeliranje kognitivnih procesa, introspekcija, eksp. psihologija)
-Sustav koji djeluje poput ljudi(Turingov test)
-Sustav koji rasuđuje racionalno(formalno, matematičko predstavljanje znanja i rasuđivanje)
-Sustav koji djeluje racionalno.
3. Što je to racionalni agent ?
racionalan agent – opaža okolinu i djeluje slijedeći svoje ciljeve temeljem stvorene slike svijeta
4. Navedi zajedničke značajke i razlike u strukturi predstavljenih racionalnih agenata.
Reakitvni agent – djeluje na temelju sadašnjeg trenutka
Reaktivno-memorijski – uz informacije sa senzora pamti i prošla stanja i odlučuje
Ciljani – traži koja mogućnost ostvaruje zadani cilj – pretražuje i planira
Korisni - traži koja mogućnost ostvaruje zadani cilj uz najveću efikasnost/zadovoljstvo
Agenti koji uče na temelju prošlih odluka
5. Koje su polazne hipoteze u oblikovanju ES utemeljenih na obradi simbola ?
- Rasuđivanje je obrada (manipulacija) simbolima
- Svaka "obrada" simbola može se izvesti na Turingovom stroju
6. Navedi razine apstrakcija u računalnom sustavu počevši od najniže razine (sklopovlje).
7. Razina znanja (Dokazivanje teorema, Prolog, OWL, …)
6. Razina simbola (LISP)
5. Algoritmi i strukture podataka, apstraktni tipovi
4. Imperativni programski jezici
3. Simbolički strojni jezik (Asemblerska razina)
2. Strojni jezik
1. Sklopovlje
7. Kojoj razini apstrakcija pripada ES implementiran u sustavu xxx (gdje xxx = LISP, sustav logike, sustav s pravilima, …)?
LISP - razina simbola
sustav logike, sustav s pravilima – razina znanja

LISP

1. Navedi razliku između imperativnih i funkcijskih programskih jezika?
-Imperativni jezici - intenzivno korištenje pridruživanja
-U čistim funkcijskim jezicima nema pridruživanja. Funkcije dohvaćaju ulazne parametre temeljem čitaj-samo,Viša razina apstrakcije,Gradbeni blokovi su izrazi koji se evaluiraju kroz primjenu funkcij
2. Navedi barem 4 značajke LISP-a.
LISP se pokreće funkcijama, **visoko rekurzivan**,**Temeljna struktura podataka je lista**, **Implicitan rad s pokazivačima,Tip varijable ne mora deklarirati unaprijed,za probleme koji nisu potpuno razumljivi**
3. Sintaksa i semantika (petlja izvođenja).

<u>Sintaksu</u> **čine simbolički izrazi**, S-izrazi su **atomi** ili l**iste, Atomi:Brojevi, Liste (S1 S2 … Sn) lista**

Semantiku određuje petlja izvođenja: čitaj-evaluiraj-ispiši svakog s-izraza.

4. Navedi rezultat izvođenja LISP koda (jednostavan primjer).

>5 -> 5 ; >() -> nil ; >(+ 1 2 3) -> 6

5. S kojom funkcijom osiguravamo lokalno vezanje simbola.

>(setq a ‘(1 2 3)) -> setq (globalno), >(let ((a 5) (b 4)) (+ a b)) -> let (lokalno)

6. Što su to lambda izrazi ?

(lambda (<formalni-parametri>) <tijelo>) - Def funkcije može se prenijeti izravno Lambda izrazom

7. Prikaži navedeni izraz točkastim parovima.

Svaka lista može se prikazati u notaciju točkastog para

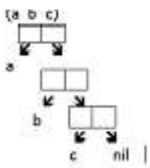
neki_atom	**=>**	**neki_atom . nil**
(a b)	**=>**	**(a . (b . nil))**
((a b) c)	**=>**	**((a . (b . nil)) . (c . nil))**

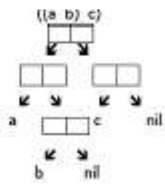
8. Kako se LISP s-izrazi (liste) smještaju u memoriji računala ?

Konstruktorske ćelije (cons cells) sadrže dva kazala (na "car" i "cdr")

9. Što obuhvaća engleski naziv “garbage collection” ?

Mehanizmi dealokacije memorije

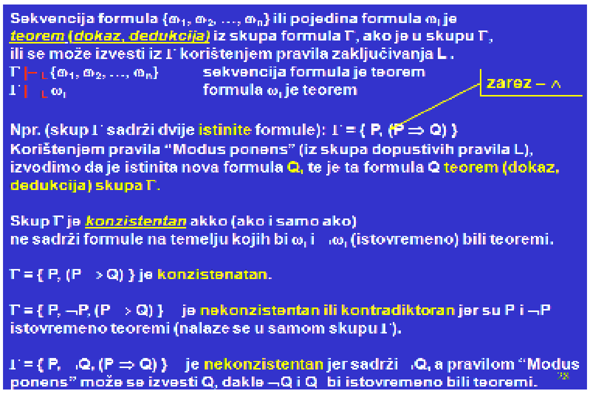


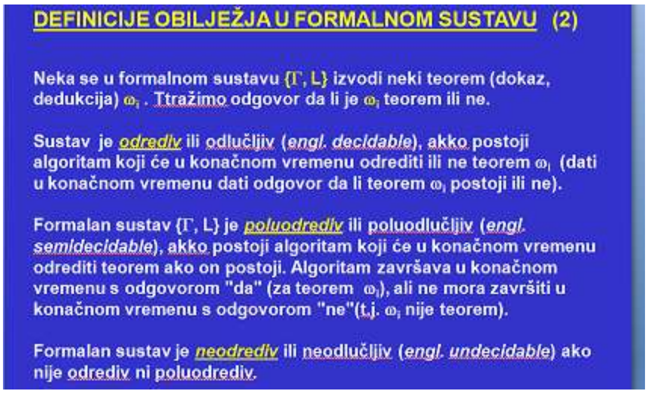


<u>Logika</u>

1. Formalan sustav, definicije temeljnih obilježja

Definiramo formalan sustav kao dvojku:{Γ, L},Γ - konačan skup ispravno definiranih formula stanje svijeta), L - skup pravila dozvoljenih





2. Primjeri nekih obilježja

P- zadovoljiva ali ne i valjana (interpretacija P=T je model, dok interpretacija P=F nije model).

(P ∨ ¬P)- valjana (tautologija), sve interpretacije (dvije) P=T, P=F, su modeli (formula je istinita).

(P ∧ ¬P)- kontradiktorna (nezadovoljiva), nema modela.

()- kontradiktorna (nezadovoljiva).

P⇒ (Q ⇒ P)- valjana (tautologija), sve interpretacije (ima ih 4: FF, FT, TF, TT) su modeli.

(P ∧ Q) - zadovoljiva. Ima samo jedan model: P=T, Q=T.

3. Preslikavanje izjavnih rečenica prirodnog jezika u dobro definirane formule predikatne logike.

∀ ide uz ⇒ , ∃ ide uz ∧ : Svi košarkaši su visoki. - ∀X ((kosarkas X) ⇒ (visok X))

4. Normalni oblici logičkih formula

Svaka propozicijska formula može se preslikati (ekvivalentna je) formuli u disjunkcijskom normalnom obliku (<u>DNF</u>) : $(k1_1 \wedge \ldots \wedge k1_n) \vee (k2_1 \wedge \ldots \wedge k2_m) \vee \ldots \vee (kp_1 \wedge \ldots \wedge kp_r)$Svaka propozicijska formula može se preslikati (ekvivalentna je) formuli u konjunkcijskom normalnom obliku (<u>CNF</u>) : $(k1_1 \vee \ldots \vee k1_n) \wedge (k2_1 \vee \ldots \vee k2_m) \wedge \ldots \wedge (kp_1 \vee \ldots \vee kp_r)$,gdje su:$k_i$ = literal (negirani ili nenegirani atomički simbol - atom),klauzula = disjunkcija literala. Npr.: $(k2_1 \vee \ldots \vee k2_m)$, CNF = konjunkcija klauzula

5. Sat problem

Tražimo model skupa formula Γ (interpretaciju koja evaluira sve formule u skupu Γ u istinito. To je ekvivalentno traženju modela jedne složene formule koja se sastoji iz konjunkcije svih formula u Γ. Iscrpna procedura rješavanja CNF SAT problema sistematski pridjeljuje istinitosne vrijednosti atomičkim propozicijskim simbolima. Za n atoma 2^n pridruživanja. Eksponencijalna složenost, računalno neizvedivo u općem slučaju. Za DNF – polinomska složenost jer postoji konačan broj literala, a dovoljno je pronaći zadovoljivost u samo jednom disjunkcijskom članu.

CNF 2SAT - polinomska kompleksnost (do 2 literala u klauzuli)

CNF 3SAT - NP kompletno (3 literala u klauzuli)

6. Teorem dedukcije

<u>Teorem</u>: Formula ψ je <u>logička posljedica</u> formule φ, t.j. $\varphi \models \psi$, akko je formula $(\varphi \Rightarrow \psi)$ tautologija (valjana). <u>Dokaz</u>: Akko je $(\varphi \Rightarrow \psi)$ tautologija (uvijek istinita), onda iz tablice za implikaciju proizlazi da kada je φ istinit (pretpostavka) i ψ mora biti istinit. To je upravo definicija logičke posljedice.

φ	ψ	$(\varphi \Rightarrow \psi)$	
F	F	T	**Budući da $(\varphi \Rightarrow \psi)$ mora bit tautologija, to njena negacija**
F	T	T	**$\neg(\varphi \Rightarrow \psi) = \neg(\neg\varphi \vee \psi) = (\varphi \wedge \neg\psi)$ mora biti nezadovoljiva. Dakle:**
T	F	F	**$\varphi \models \psi$ akko je $(\varphi \wedge \neg\psi)$ <u>nezadovoljiva, t.j. kontradiktorna</u>**
T	T	T	

7. Pravilo razrješavanja (Robinson) i svođenje elementarnih pravila na razrješavanje

Logička posljedica dviju istinitih, konjunkcijom vezanih, univerzalno kvantificiranih normaliziranih klauzula je normalizirana klauzula bez <u>jednog</u> komplementarnog para literala. Normalizirana klauzula je formula oblika disjunkcije literala. Komplementarni parovi su nenegirani i negirani jednaki atomički predikati. Robinsonov teorem zadržava zadovoljivost rezultirajuće formule.

1. $(A_1 \vee A_2 \vee \ldots \vee A_n \vee B)$ 2. $((\neg B) \vee C_1 \vee C_2 \vee \ldots \vee C_p)$ -> 3. $(A_1 \vee A_2 \vee \ldots \vee A_n \vee C_1 \vee C_2 \vee \ldots \vee C_p)$

3 - razriješena formula (engl. resolvent), bez kompl. para B i $\neg B$.

<u>Primjer:</u> $(P \vee Q)$ Klasičan Izvod: $(P \vee Q) \wedge (P \vee \neg Q) = P \vee (Q \wedge \neg Q) = P$

$\underline{(P \vee \neg Q)}$ uporabom distribucije: $A \vee (B \wedge C) = (A \vee B) \wedge (A \vee C)$

$(P \vee P) = P$

<u>Modus ponens</u> je	**P**	**P**
poseban slučaj razrješavanja:	**$\underline{(P \Rightarrow Q)}$**	**$\underline{(\neg P \vee Q)}$**
	Q	**Q**
<u>Modus tolens</u> je	**$\neg Q$**	**$\neg Q$**
poseban slučaj razrješavanja:	**$\underline{(P \Rightarrow Q)}$**	**$\underline{(\neg P \vee Q)}$**
	$\neg P$	**$\neg P$**
<u>Lančano pravilo</u> je	**$(P \Rightarrow Q)$**	**$(\neg P \vee Q)$**
poseban slučaj razrješavanja:	**$\underline{(Q \Rightarrow R)}$**	**$\underline{(\neg Q \vee R)}$**
	$(P \Rightarrow R)$	**$(\neg P \vee R) \equiv (P \Rightarrow R)$**

8. Normalizirane klauzule

Normalizirana klauzula: disjunkcija literala.
Temeljna ili osnovna klauzula (engl. ground clause): klauzula bez varijabli.Preslikavanje ne zadržava jednakost (ekvivalenciju),ali zadržava zadovoljivost (dostatno za uporabu teorema oborivosti).

9. Izjednačavanje atomičkih predikata

U razrješavanju tražimo komplementarne parove (radi eliminacije).
U predikatskoj logici jedinični predikati s istim predikatskim simbolom i jednakim brojem članova možda se mogu izjednačiti. Iako izvorno nejednaki, treba pronaći supstituciju varijabli, koja bi ih učinila jednakima. Doseg varijabli je unutar jedne klauzule, te svaku supstituciju varijable treba dosljedno provesti u cijeloj klauzuli.

<u>Primjer 1:</u>
1. (p1 a b) ; p1 je pred. simbol, a i b se konst.
2. ((¬(p1 X b)) ∨ (p2 X c)) ; p1 i p2 su pred., X je varijabla
supstitucija: X=a u cijeloj klauzuli 2.
1. (p1 a b) 2. ((¬(p1 a b)) ∨ (p2 a c))
(…) su kompl. parovi koji se u razrješavanju eliminiraju. Ostaje samo (p2 a c).
Skup parova varijabli i pripadnih supstitucija je unifikator: $\beta=\{X=a\}$.

10. Navedi algoritam za opći postupak dokazivanja teorema

1.definiraj skup aksioma E kao ispravne formule (wff) = stanje svijeta.
2.definiraj teorem H kao ispravnu formulu (wff).
3.negirani H dodaj skupu aksioma E, dobiven je skup formula F.
4.preslikaj skup formula F u skup univerzalno kvantificiranih normaliziranih klauzula K.
Zatim ponavljaj: izaberi klauzule k1 i k2 iz skupa K
novi-k := razriješi (k1, k2)
dodaj novi-k u skup normaliziranih klauzula K
novi-k' := pojednostavi novi-k sa K ; pojednost. novi-k
K := pojednostavi K sa novi-k' ; pojednostavi skup K
dok: () ili redukcija više nije moguća;ako (), vrati "teorem H dokazan"
inače, vrati "skup konzistentan, teorem H nije dokazan"
Važno je uočiti da u svakom prolazu kroz petlju novi-k proširuje skup klauzula (dodaje se u skup a sve izvorne klauzule ostaju u skupu).Skup se zatim postupcima pojednostavljivanja nastoji smanjiti, najčešće bezuspješno.

11. Strategije u razrješavanju

1. <u>Strategija izbora u širinu</u> (*engl. breadth-first-strategy*)
U prvoj iteraciji svaka klauzula se uspoređuje sa svakom klauzulom.
U n-toj iteraciji, novo generirane klauzule iz n-1 iteracije dodaju se skupu klauzula te se sve klauzula uspoređuju svaka sa svakom.
Generira eksponencijalan broj novih klauzula.
Garantira pronalaženje rješenja (strategija je kompletna).
2. <u>Strategija s prvenstvom jediničnih klauzula </u>(jedan literal)
Cilj dokazivanje je prazna klauzula.
Stoga u svako koraku ako se proizvede klauzula s manje članova od izvornih dolazi se bliže rješenju
Razrješavanje s klauzulom koja sadrži samo jedan literal sigurno generira kaluzulu s manje članova nego što ih imaju polazne.
3. <u>Strategija skupa potpore</u>
Za skup ulaznih klauzula K, može se definirati podskup $T \subset K$, (skup potpore).

Ova strategija zahtijeva da barem jedna od klauzula u razrješavanju bude ili ima pretka u T.
Može se pokazati da ako je K nezadovoljiv, a K-T zadovoljiv, strategija skupa potpore je kompletna.
Ako je izvorni set klauzula-aksioma E_i zadovoljiv (uobičajen slučaj), nije oportuno tražiti razrješavanje u tom skupu (K-T skup)Negacija onoga što nastojimo dokazati daje nekonzistentnost.
Strategija skupa potpore zahtijeva stoga da barem jedna od klauzula u razrješavanju bude negirani zaključak ili klauzula generirana ranije razrješavanjem s negiranim zaključkom.
Reducira prostor stanja strategije izbora u širinu, te traži samo sva razrješavanja s negiranim zaključkom i njegovim potomcima, a ne sva kao u strategiji u širinu.
4. <u>Linearna ulazna strategija razrješavanja (PROLOG)</u>
Polazi od negiranog zaključka i jedne klauzule iz skupa aksioma.
U svakom slijedećem koraku razrješava se uvijek s rezultatom iz prijašnjeg koraka i jednom novom klauzulom iz skupa aksioma.
Nikad se ne koriste klauzule generirane u nekom od ranijih koraka (kao kod npr. strategije skupa potpore), niti dva aksioma zajedno.
Linarna ulazna strategija nije kompletna.
12. Pojednostavljivanje u skupu formula predikatne logike
1. Uporaba temeljnih pravila: $(P \vee R \vee P)$ pojednostavi (spoji) u $(P \vee R)$
$(P \vee \neg P \vee Q)$ izostavi cijelu jer je evidentno valjana (T)
2. Podrazumijevanje <u>klauzula(odbacuje manje općenite klauzule)</u> Klauzula ω_1 podrazumijeva klauzulu ω_2 , ako su literali u ω_1 podskup literala u ω_2.$(P \vee R)$ podrazumijeva klauzule $(P \vee R \vee Q)$ ->Svaka disjunkcija s istinitom formulom je istinita.
13. Goedelov teorem nekompletnosti
U FOPL se uvodi skup prirodnih brojeva (N) i aritmetika.
"Ne postoji konzistentan i kompletan sustav dokazivanja za FOPL + N."
Postoje izrazi koji su: istiniti ali nedokazljivi, ili dokazljivi ali nisu istiniti.
Aritmetika omogućuje izgradnju kodnog sustava (Godelovi brojevi) za izraze u FOPL + N:
Npr.: P = " P je nedokazljiv"
Ako je P istinit, tada je P nedokazljiv (nekompletnost !).
Ako je P neistinit, tada je P dokazljiv (nekonzistentnost !).

<u>Pravila</u>
1. Koja je temeljna razlika između sustava s pravilima i formalne logike u modeliranju znanja ?
Namjera materijalne implikacije je modelirati <u>uvjetnu konstrukciju,</u> (<u>a ne uzročno-posljedičnu vezu</u>) – pojam implikacije (sustav s pravilima ako onda bez obzira na logiku)
2. Da li sustavi s pravilima imaju proceduralan upravljački tijek ili neproceduralan, te zašto ?
Pravila koja postignu slaganje se izvode (<u>neproceduralan</u> tijek).
3. Navedi barem tri potencijalne poteškoće u radu sustava s pravilima.
Činjenice mogu biti složene strukture podataka (usporava poklapanje).
Konjunkcija u AKO stranama pravila (produljeno vrijeme poklapanja).
Slaganje činjenica s više pravila (mogućnost konflikta).
Povezani ciklusi zaključivanja (mogućnost ulaska u beskonačnu petlju).
4. Koja su dva temeljna postupka zaključiva u sust s pravilima. Koncizno opiši izvođenje oba postupka
<u>Zaključivanje ulančavanjem unatrag</u>
Neka postoji činjenica P(a) i pravilo: $(P(X) \rightarrow Q(X))$. <u>X je varijabla</u>. P(a) ; $(P(X) \rightarrow Q(X))$
<u>Korisnik postavlja upit ili cilj</u> koji se pretražuje u činjenicama, pa ako se ne nalazi nastoji se potražiti

u TADA stranama čija se potvrda (novi cilj) traži u AKO stranama istih pravila i u činjenicama:
? P(a) - ovdje odmah odgovor DA.
? R(b) - nema u činjenicama, a također je i TADA strana pravila različita, odgovor je NE.
? Q(a) - nema u činjenicama, ali se odgovarajućom supstitucijom varijabli (X = a) može pronaći u TADA strani pravila.
Supstitucija se mora provesti u cijelom pravilu. Novo stanje sustava je:P(a); (P(a) → Q(a))
Potvrda za Q(a) se treba potražiti u AKO strani, tj., <u>novi cilj je P(a)/</u>P(a) se nalazi u činjenicama.
Time je istinitost Q(a) potvrđena te se ta nova činjenica najčešće dodaje u skup činjenica kako bi u mogućem budućem upitu odgovor bio brži. Novo stanje sustava je : Q(a);P(a);(P(a) → Q(a))
Konačno, varijable u pravilima se oslobađaju i sus je spreman za nov ciklus: Q(a);P(a);(P(X) → Q(X))
Zaključak je uvijek unija odgovora i provedene supstitucije!
Ciljevi i podciljevi mogu biti višestruki povezani konjunkcijom!
<u>Zaključivanje ulančavanjem unaprijed</u>
Neka u sustavu postoji pravilo (P(X) → Q(X)) ;U sustav se zatim, kao rezultat nekog događaja, upiše činjenica P(a), te je stanje : P(a) / (P(X) → Q(X))
U trenutku upisa P(a) u sustav, započinje automatska (samostalna) potraga za pravilom koje u svom AKO dijelu ima strukturu koja bi se mogla izjednačiti s P(a). Supstitucijom X=a to se nalazi, pa je stanje P(a) / (P(a) → Q(a)). U pravilu je zadovoljen uvjet (AKO), te se posljedica izvođenja pravila upisuje u sustav kao nova činjenica Q(a). Sada je stanje : Q(a);P(a);(P(a) → Q(a))
Konačno varijable se oslobađaju sustav je spreman za novi ciklus u stanju: Q(a);P(a);(P(X) → Q(X))
5. Što je to "konfliktni skup" (u sustavu CLIPS "Agenda") ?
Kada stanje činjenica zadovoljava više pravila te je potrebno izabrati jedno iz skupa.
6. Kada treba odabrati ulančavanje unaprijed a kada unatrag ? (objasni posebno za slučaj moguće analize pravila, a posebno za slučaj analize činjenica).
1. <u>Analiziraju se AKO i TADA strane pravila.</u>
Ukoliko postoji mnogo uvjeta (više nego zaključaka) u "tipičnom pravilu", to ukazuje na potrebu ulančavanja unaprijed.
Ulančavanje unatrag generiralo bi mnogo novih upita.
Ukoliko postoji malo uvjeta: ulančavanje unatrag.
2. Ako nije moguće zaključiti dominantnost u pravilima potrebna je <u>analiza skupa činjenica</u>
Ako su sve činjenice poznate, a potrebno je vidjeti kamo to vodi, koristi se ulančavanje unaprijed.
Ako je malo činjenica poznato a cilj je potvdti jdnu od mnogih hipoteza koristi ulančavanje unatrag
7. Nabroj četiri jezgrena modula u klasičnom ES temeljenom na pravilima.
skup pravila(bza znanja);skup činjenica(radna memorija);stroj za zaključivanje; sučelje s korisnikom
8. Nabroji barem pet prednosti sustava s pravilima.
Izravni prirodni prikaz i korištenje empiričkog znanja prikupljenog od eksperata određene (uske) domene. ; Modularnost zbirke znanja i njeno jednostavno održavanje. ;Efikasno izvođenje u uskim domenama. ;Dobra obilježja razjašnjavanja procesa automatiziranog rasuđivanja (kako, zašto). ; Skup pravila prirodno se preslikava u problem pretraživanja prostora stanja. ;Koraci tijekom rješavanja problema su transparentni. Olakšava izradu sučelja temeljenih na dijalogu s korisnikom.
9. Nabroji barem četiri nedostatka sustava s pravilima.
Sustav s pravilima naglo prestaje biti uporabiv na rubu domene, umjesto da poput eksperata monotono slabi u rješavanju problema.; Razjašnjenja, koja slijede iz prikaza ulančavanja, ne daju kauzalne veze u procesu (nema povezivanja dubinskih uzroka i posljedica).;
Znanje prikazano pravilima prilagođeno je određenom zadatku, s malom fleksibilnošću i širinom –

pretpostavka zatvorenog svijeta (što je vrlo različito prema znanju eksperata).;U razvoju sustava postoji poteškoća razumijevanja između inženjera znanja i eksperta domene.

Neizrazita logika i teorija mogućnosti

1. Definicije neizrazitih (lingvističkih) varijabli

Skup je neizrazit ako karakteristična funkcija može poprimiti bilo koju vrijednost iz intervala [0, 1]
Lingvistička varijabla opisana je vrijednost karakteristične funkcije (diskretnom ili kontinuiranom)

2. Kako se definiraju logičke operacije u neizrazitoj logici ?

Svaka atomička izjava preslikava se u karakterističnu funkciju (diskretnu ili kontinuiranu).
Logički operatori (izvode se nad vrijednostima karakterističnih funkcija i rezultiraju u novoj karakterističnoj funkciji):

Negacija: $\mu(\neg A) = 1 - \mu(A)$

Konjunkcija: $\mu(A \wedge B) = \min[\mu(A), \mu(B)]$

Disjunkcija: $\mu(A \vee B) = \max[\mu(A), \mu(B)]$

$\mu(A \rightarrow B) = \max[(1 - \mu(A)), \mu(B)] = 1 - \min[\mu(A), (1 - \mu(B))]$

3. Što je to "inženjerska" definicija implikacije u neizrazitoj logici ?

$\mu(A \rightarrow B) = \min[\mu(A), \mu(B)]$ **ili** $\mu(A \rightarrow B) = [\mu(A) \cdot \mu(B)]$ **(jednostavnije)**

4. Kako se zaključuje (postupak izvođenja zaključka) u poopćenom Modus ponensu ?

A = {1/1, 2/0.6, 3/0.2, 4/0}

B: karakterističnu funkciju definiramo kao matricu koja ima vrijednosti 1 ako su vrijednosti x i y jednake, 0.5 ako su susjedne (malo se razlikuju) i 0 ako su udaljene.

$\mu(A) \circ \mu(B)$ = max { min [(1 , 1), (0.6, 0.5), (0.2, 0), (0, 0)], min [(1, 0.5), (0.6, 1), (0.2, 0.5), (0, 0)],min [(1, 0), (0.6, 0.5), (0.2, 1), (0, 0.5)],min [(1, 0), (0.6, 0), (0.2, 0.5), (0, 1)] } = = max { [1, 0.5, 0, 0],[0.5, 0.6, 0.2, 0],[0, 0.5, 0.2, 0],[0, 0, 0.2, 0] } = = {1, 0.6, 0.5, 0.2} – sličan x pretpostavka potvrđena

5. Objasni postupak zaključivanja s neizrazitim pravilima:

- Činjenice izrazite, AKO strana pravila neizrazita, TADA strana pravila izrazita.
-Činjenice izrazite, AKO strana pravila neizrazita, TADA strana pravila neizrazita
-Činjenice neizrazite, AKO strana pravila neizrazita, TADA strana pravila izrazita
-Činjenice neizrazite, AKO strana pravila neizrazita, TADA strana pravila neizrazita

6. Navedi dva postupka kompozicije svih pravila u izvođenju nezrazitog zaključka

kompozicija svih pravila MAX postupkom ; kompozicija svih pravila SUMIRANJEM

7. Sustav s izrazitim činjenicama koje imaju faktore izvjesnosti i izrazitim pravilima koja imaju faktore izvjesnosti - Kako se računa faktor izvjesnosti zaključka jednog pravila?

Računanje s faktorima izvjesnosti CF

- Izvorni faktori izvjesnosti činjenica CF_i dobiveni **heuristički**.
- Izvorni faktori izvjesnosti pravila CF_R dobiveni **pojednostavljenom uvjetnom vjerojatnosti**.

1. Određivanje CF s izrazitim vrijednostima

Činjenice: izrazite s faktorima izvjesnosti CF1 . . . CFn
Pravilo: AKO strane izrazite, TADA strana izrazita ili ne, faktor izvjesnosti pravila CF_R .

Novi faktor izvjesnosti pravila:
$CF_C = CF_R \cdot \min[CF1 , . . . , CFn]$

Odabere se minimalni CF temeljem slaganja parova činjenica s AKO dijelovima u pravilu. Dobiveni minimalni CF množi izvorni CF_R pravila. Slijedi faktor izvjesnosti zaključka (TADA strane pravila) CF_C .

Moramo promotriti zasebno svaki par prekrivanja (jedna neizrazita činjenica i jedan neizraziti dio u AKO strani pravila).

Činjenica: jedna neizrazita, faktor izvjesnosti $CF^{\sim}_1$ (npr. 0.8)
Pravilo: AKO strana neizrazita, TADA strana izrazita ili ne, faktor izvjesnosti pravila $CF^{\sim}_R$ (npr. 0.7)
Novi faktor izvjesnosti pravila:
$CF_C = (CF^{\sim} \cdot S_1) \cdot CF^{\sim}_R$
gdje je S_1 = mjera sličnosti činjenice i AKO dijela s kojim se činjenica prekriva.

Ako postoji prekrivanje **više parova činjenica i dijelova AKO strane**, u izračunu se uzima **minimalni** umnožak faktora izvjesnosti činjenice i odgovarajuće sličnosti s AKO dijelom pravila. Taj se umnožak dalje množi s izvornim faktorom izvjesnosti pravila $CF^{\sim}_R$.
$CF_C = \{\min[(CF_1 \cdot S_1), (CF_2 \cdot S_2), \ldots (CF_m \cdot S_m)] \cdot CF^{\sim}_R \}$

8. Sustav s neizrazitim činjenicama koje imaju faktore izvjesnosti i neizrazitim pravilima koja imaju faktore izvjesnosti - Kako se računa faktor izvjesnosti zaključka jednog pravila? (gore odg.)

9. Objasni riječima što je to "mjera sličnosti" u eksp sust s neizrazitim pravilima i kako se upotrebljava

S = P(R | F) ako N(R | F) > 0.5

= [N(R | F) + 0.5] · P(R | F) inače; gdje je:

R = μR = karakt. funkcija jednog konjunkcijskog dijela AKO strane

F = μF = karakt. funkcija jedne činjenice

P(R | F) = max [min (μR, μF)] - mjera mogućnosti (engl. possibility)

N(R | F) = 1 - P(R' | F) - mjera nužnosti (engl. necessity) R' je komplement, t.j. μR' = 1 - μR

10. U ekspertnom sustavu s faktorima izvjesnosti (činjenica i pravila), kako se računa faktor izvjesnosti činjenice koja se temeljem aktivacije nekog pravila upisuje u skup činjenica?

Ako se aktivacijom pravila upisuje nova činjenica, njen izvorni faktor izvjesnosti se modificira tako da se množi sa skupnim faktorom AKO strane i s faktorom izvjesnosti pravila.

11. Kako se izračunava višestruki doprinos jednoj hipotezi (višestruko upisivanje iste činjenice (izrazite ili neizrazite)?

Pravilo može svojim zaključkom upisati činjenicu F_n (uz faktor izvjesnosti CF_{Fn}), koja već postoji u skupu činjenica kao F (uz faktor izvjesnosti (CF_F).
(Vidi raniji primjer uz pretpostavku da je F1 = F2).

Obnovljena (ista) činjenica dobiva novifaktor izvjesnosti:
$CF = max [CF_{Fn} , CF_F]$

- Ako su nova i stara vrijednost činjenice izrazite: obnovljena izrazita **vrijednost ostaje neizmijenjena**. Mijenja se samo faktor izvjesnosti (max).
- Ako su nova i stara vrijednost činjenica **neizrazite**: obnovljena **vrijednost** je **unija dviju karakterističnih funkcija** (karakteristične funkcije iste neizrazite činjenice mogu se razlikovati zbog npr. primjene različitog modifikatora lingvističke varijable – činjenice).

Bayesove mreže

1. Definicije: slučajna varijabla, uvjetna vjerojatnost, marginalan razdioba, pravilo zbrajanja, lančano pravilo, Bayesovo pravilo

- Vjerojatnost je broj u intervalu [0, 1] koji izražava izglednost nastupanja nekog događaja

- <u>Slučajna varijabla</u> je varijabla koja može poprimiti vrijednosti iz skupa isključivih i potpunih vrijednosti (prostora vrijednosti ili stanja) s <u>određenom vjerojatnošću</u>.

- P(A | B) = vjerojatnost da će se dogoditi A ako se dogodio B. ; P(A | B) = P(A $\cap$ B) / P(B) ili uobičajeno: P(A | B) = P(A, B) / P(B) gdje je: P(A, B) vjer. istodobnog pojavljivanja A i B P(B) vjer. pojavljivanja B ; P(A | B) = 1 je analogno B $\rightarrow$ A (implikacija)

Ako su A i B diskretne binarne slučajne varijable uz $\Omega_{A,B}$ = {T, F}, računamo npr.:

P(A=T | B=T) = P(A=T $\wedge$ B=T) / P(B=T)

- Marginalna je razdioba nekog fiksiranog podskupa vrijednosti tih varijabli, odnosno razdioba toga podskupa bez obzira na vrijednsti u ostatku skupa-P(A,B) = $\sum_c$ P(A=a, B=b, C=c)-vjer da će A=a i B=b

-<u>pravilo zbrajanja:</u> Neka je dana zajednička razdioba dviju diskretnih binarnih varijabli: P(A, B). Marginalna razdioba u odnosu na A (vjer. da A=a neovisno o B) iznosi: P(A=a) = $\sum_b$ P(A=a, B=b) Kako iz definicije uvjetne vjerojatnosti slijedi: P(A=a, B=b) = P(A=a | B=b) P(B=b) To marginalnu razdiobu možemo računati preko uvjetne vjerojatnosti: P(A=a) = $\sum_b$ P(A=a | B=b) P(B=b)

-<u>bajesov</u> Iz definicije uvjetne vjerojatnosti slijedi:P(A | B) = P(A, B) / P(B) a također i: P(B | A) = P(A,

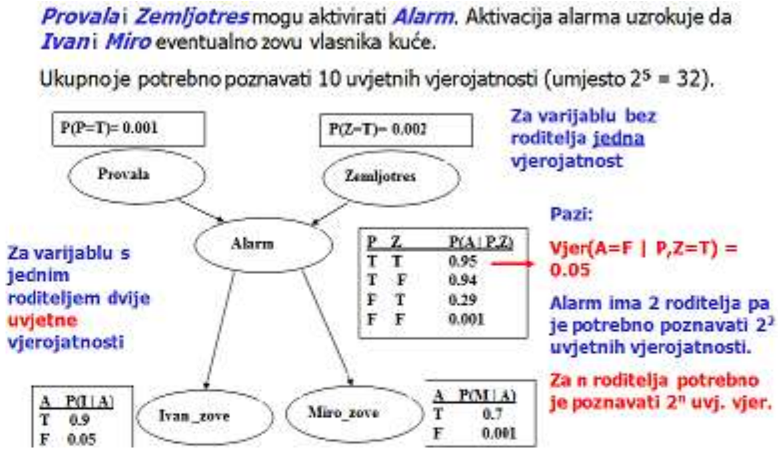
B) / P(A) Ako izraze izjednačimo po zajedničkom članu P(A, B) slijedi: P(A | B) = (P(B | A) P(A))/P(B)

- <u>Lančano pravilo : P(X1, … , Xn) = $\prod_i$ P(Xi | Xn-1, …, X1)</u>

2. Što je to Markovljeva uvjetna nezavisnost (možeš opisati riječima)?

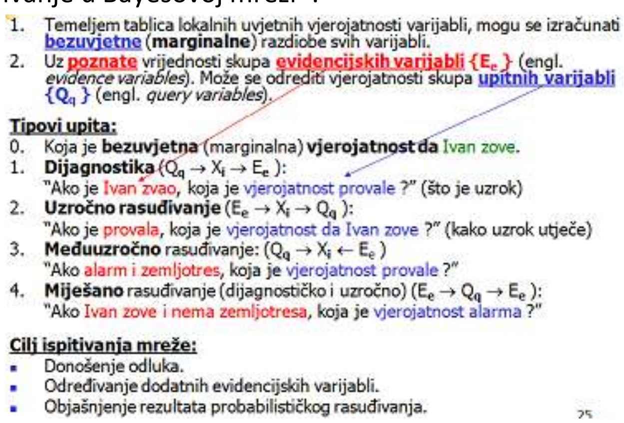
3. Navedi postupak oblikovanja Bayesove mreže.

1.Odaberi skup slučajnih varijabli koje opisuju domenu problema. 2.Uredi skup varijabli (povlačenjem lukova) tako da najprije odrediš najranije prethodnike (varijable koje nemaju roditelja), a zatim varijable na koje one izravno utječu (neposredne lokalne uzročne veze). 3.Ponavljaj postupak pod točkom 2 do krajnjih varijabli (djece).4.Definiraj tablice lokalnih uvjetnih vjerojatnosti svake varijable (vjerojatnosti te varijable uz uvjet da njeni roditelji zauzmu svoje vrijednosti). Pri tome broj roditelja neke varijable određuje dim. njene lokalne tablice vjerojatnosti U slučaju diskretnih binarnih varijabli za *m* roditelja potrebno je poznavati 2^m vjerojatnosti.



4. Zašto u Bayesoj mreži nije potrebno poznavati svih 2^n vjerojatnosti za ***n*** Booleovih slučajnih varijabli koje opisuju neku domenu? **(odgovor pod 2)**

5. Što je to "rasuđivanje u Bayesovoj mreži"?



6. Za elementarnu Bayesovu mrežu sastavljenu od dvije Booleove slučajne varijable izračunaj:

- Bezuvjetne (marginalne) razdiobe vjerojatnosti -> **Mreža je definirana jednim lukom i uvjetnim vjerojatnostima: P(A=T) = … P(B=T | A=T) = … P(B=T | A=F) = …**

Čvor A: P(A=T) = … (to je bilo odmah zadano)

Čvor B: P(B=T) = $\sum_a$ P(B=T | A=a) P(A=a) a = {T, F}

Računamo marginalnu razdiobu za fiksirani B=T.Sumacija se obavlja po vrijednostima za čvor A

- Uzročno djelovanje

P(A=T) = 1, t.j. ako je poznato (vidljivo, evidentno) stanje A, kako to mijenja naše vjerovanje u B ? Poznato P(A=T) je uzrok koji propagira do B. P(B=T | A=T) = ?, Vrijednost možemo očitati izravno iz zadane tablice. Ta vjerojatnost je različita od ranije izračunate bezuvjetne vjerojatnosti. Saznanjem da je A=T primijenilo se naše vjerovanje u B.

- Dijagnostičko djelovanje -> **(B šalje poruku prema A, da je B u T)**

P(B=T) = 1, t.j. ako je poznato (vidljivo, evidentno) stanje B, kako to mijenja naše vjerovanje u A ? P(A=T | B=T) = ? Bayes: P(A=T | B=T) = P(B=T | A=T) P(A=T) / P(B=T)

P(B=T | A=T) i P(A=T) očitamo iz zadanih tablica uvj. vjerojatnosti. Za nazivnik: P(B=T) = $\sum_a$ P(B=T, A=a) = Σ_a P(B=T | A=a) P(A=a) računamo <u>marginalnu razdiobu</u> pravilom zbrajanja. Saznanje da je B=T mijenja naše ranije bezuvjetno vjerovanje u A.

Neka je zadano početnih pet vjerojatnosti:

P(O=T) = 0.4	[P(O=F) = 0.6]
P(L=T | O=T) = 0.6	[P(L=F | O=T) = 0.4]
P(L=T | O=F) = 0.1	[P(L=F | O=F) = 0.9]
P(C=T | O=T) = 0.8	[P(C=F | O=T) = 0.2]
P(C=T | O=F) = 0.3	[P(C=F | O=F) = 0.7]

Zanima nas vjerojatnost **P(C=T | L=T)**, t.j. kako se promijenilo naše vjerovanje u P(C=T) kada smo opazili L=T (evidencija).

Prvo računamo **bezuvjetnu** (**marginalnu**) vjerojatnost čvora C, t.j. **P(C=T)**, (sumiramo po svim vrijednostima O=o, **ne ovisi o L**):

P(C=T) = Σ_o P(C=T | O=o) P(O=o) = pravilo zbrajanja
= P(C=T | O=T) P(O=T) + P(C=T | O=F) P(O=F)
= (0.8 x 0.4) + (0.3 x 0.6) = **<u>0.5</u>**

Neka sad opazimo da je L=T.
L šalje dijagnostičku potporu u čvor O.
Računamo novu vjerojatnost za čvor O, t.j. P(O | L=T):

P(O=T | L=T) = Bayes
= P(L=T | O=T) P(O=T) / $\underbrace{\Sigma_o \text{ P(L=T | O=o) P(O=o)}}_{\text{P(L=T) – marginalna razdioba}}$ =
= (0.6 x 0.4) / [(0.6 x 0.4) + (0.1 x 0.6)]
= 0.24 / 0.3 = 0.8

Također:
P(O=F | L=T) = 0.2

Nakon toga, čvor O šalje uzročnu potporu u čvor C.
Umjesto ranijih P(O=T) i P(O=F) u bezuvjetnoj (marginalnoj) razdiobi za P(C=T), sada, koristimo: P(O=T | L=T) i P(O=F | L=T).

P(C=T | L=T) = Σ_o P(C=T | O=o) **P(O=o | L=T)** =
= P(C=T | O=T) P(O=T | L=T) + P(C=T | O=F) P(O=F | L=T)
= (0.8 x 0.8) + (0.3 x 0.2) = 0.64 + 0.06 = **<u>0.7</u>**

Zaključak:
Nakon evidencije L=T, vjerovanje u C promijenilo se od početnih 0.5 na 0.7.

7. Za koji oblik Bayesove mreže je moguća polinomska složenost rasuđivanja ?

Za topologiju <u>jednostruko povezanih čvorova</u> (polistablo), gdje postoji najviše jedan put između dva čvora (uvijek samo jedan roditelj), moguće je oblikovati točan proračunski algoritam polinomske složenosti.