UNIVERZITET CRNE GORE

Elektrotehnički fakultet, Podgorica

Materijal sa osmog termina predavanja iz

EKSPERTNIH SISTEMA

Predstavljanje znanja u formalnoj logici

Prof. dr Vesna Popović-Bugarin

Podgorica, 2018.

Predstavljanje znanja u formalnoj logici

3.1 Predstavljanje znanja

Prilikom odlučivanja da li je za rješavanje nekog problema pogodno koristiti ekspertne sisteme ključni faktor, pored složenosti i širine postavke zadatka, je i karakter zadatka koji je potrebno riješiti. Karakter mora biti takav da se rješavanje zadatka izvodi manipulacijom simbolima i simboličkim strukturama, što je i osnovna razlika od klasičnog programiranja. U ovom poglavlju će bit dat kratak pregled nekoliko najpopularnijih metoda za predstavljanje znanja (reprezentaciju stanja svijeta i znanja vezanih za posmatrani domen u simboličkom obliku) i pridruženih tehnika rezonovanja (rasuđivanja na osnovu činjenica) kojima će se izvoditi nova znanja o svijetu i ista koristiti kako bi se odlučilo koja akcija da se preduzme ili koji odgovor na postavljeno pitanje da se da.

Znanje omogućava ekspertnom sistemu da uspješno i usmjereno bira akcije koje će ga dovesti do optimalnog rješenja. Znanje u ekspertnim sistemima igra naročito značajnu ulogu kod djelimično osmotrivih stanja svijeta. U ovakvim situacijama se generalno znanje – znanje koje je opšte važeće za posmatrani problem kombinuje sa činjenicama koje se mogu zapaziti posmatranjem stanja svijeta (postavljanjem pitanja korisniku) kako bi se zaključilo koji su skriveni aspekti trenutnog stanja svijeta, prije nego što se odluči na koji način treba djelovati. Na primjer, kada ljekar liječi pacijenta, on kombinuje opšta pravila koja su vezana za neka oboljena, koja je naučio u toku školovanja i na osnovu iskustva, sa simptomima koje pacijent ima, kako bi zaključio kakvo je stanje oboljenja pacijenta koje nije direktno vidljivo.

Da bi se moglo govoriti o načinima predstavljanja znanja, najprije je neophodno definisati pojam znanja. Kada je u pitanju ekspertni sistem znanje se definiše kao razumijevanje predmetne oblasti (domena) analogno znanju ljudskog eksperta. Međutim, da bi se znanje koristilo u ekspertnom sistemu, potrebno je nekako njegove elemente kodirati i smjestiti u ekspertni sistem – predstaviti znanje na način razumljiv računaru. **Predstavljanje znanja** je metod kodiranja znanja u bazi znanja nekog ekspertnog sistema. Predstavljanje znanja obuhvata:

- strukturu (model) koja se zahtijeva da opiše elemente znanja (formalna logika, predikatska logika, semantičke mreže, okviri, pravila),
- interpretativni proces koji se zahtijeva u korišćenju znanja – rezonovanju na osnovu znanja.

Neformalno se može reći da je baza znanja sastavljena od rečenica. Ovdje se misli na tehničko značenje pojma rečenica, koje nije isto kao značenje termina rečenica u prirodnom jeziku. Svaka rečenica je izražena u jeziku koji se naziva **jezik predstavljanja znanja**, i kojim se predstavlja neka tvrdnja [2]. Rečenicama koje se koriste u ekspertnim sistemima se izražava ne samo neko naučno tvrđenje: naučna činjenica, hipoteza, naučni zakon, već i svjedočenje koje se navodi za to tvrđenje kako bi se na osnovu činjenica koje se mogu osmotriti zaključilo da li to tvrđenje važi u konkretnom slučaju. Osim toga, rečenicama se pored smisla može pripisati jedno veoma bitno svojstvo: istinitost ili lažnost.

Pod rečenicom se, uopšteno govoreći, podrazumijeva takva kombinacija riječi i/ili znakova nekog (prirodnog ili vještačkog) jezika, izgrađena prema pravilima toga jezika, koja može da bude upotrijebljena za tvrđenja, postavljanje pitanja, izražavanje sumnji, izdavanje naloga, ukoravanje, itd. i koja mora imati smisao. Primjeri rečenica su:

1. Da li ste provjerili ove podatke?
2. Treba ponoviti ovo ispitivanje.
3. Uran je radioaktivni hemijski element.

Jasno je da se u ekspertnom sistemu ne mogu pojaviti sva tri navedena tipa rečenica. O prve dvije rečenice se ne može govoriti kao o istinitim, odnosno lažnim, bar ne onako kao o trećoj rečenici. Ovakve rečenice se mogu pojaviti samo u odgovoru koji daje ekspertni sistem, a nikako kao činjenice koje čine bazu znanja. Rečenice koje izražavaju pitanja, sumnju, savjete, zapovijesti, prekore, nadanja, bojazni, itd. ništa izričito ne tvrde, pa se ne postavlja pitanje njihove istinitosti, odnosno, lažnosti i ne mogu predstavljati činjenice koje se nalaze u bazi znanja. O trećoj rečenici se može misliti i govoriti kao o istinitoj i može se smatrati tehničkom rečenicom.

Iskazi su rečenice o kojima se može misliti (i govoriti) kao o istinitim, (odnosno lažnim). Za označavanje smisla rečenice o kojoj se može misliti kao o istinitoj ili lažnoj, u logici se koristi izraz **sud**.

Mora postojati način da se u bazu znanja dodaju nove rečenice, kao i način da se ne ponavljaju iste rečenice (vodi računa o onome što se već zna). Dodavanje novih rečenica u bazu znanja se vrši ili postavljanjem pitanja korisniku i smještanjem njegovih odgovora, smještanjem zapaženih novih činjenica, ali isto tako i rezonovanjem na osnovu trenutnog stanja svijeta i opštih znanja koja se nalaze u BZ i izvođenjem novih rečenica – znanja na osnovu postojećih. Dakle, od ekspertnog sistema se očekuje da na postavljeno pitanje može dati odgovor i kada se on ne nalazi eksplicitno u BZ, ali slijedi na osnovu znanja koja su smještena u njoj. Baza znanja se na početku može sastojati od nekog opšteg znanja za posmatrani problem koje je prikupljeno u toku njenog projektovanja. Prilikom svakog poziva programa – korišćenja ekspertnog sistema on prvo opaža trenutno stanje svijeta, bilo postavljanjem nekih pitanja ili korišćenjem senzora koje ima, dopunjava BZ ovim novim opažajima i koristi tu dopunjenu BZ kako bi zaključio koja akcija da se odabere. Kada se odabrana akcija i izvrši BZ se ažurira kako bi oslikala novonastalo stanje svijeta.

3.1.1 Tipovi znanja

Znanje se može razvrstati na više različitih načina. Samim tim, ne postoji idealan tip predstavljanja znanja koji je pogodan za sve primjene. Naime, različiti tipovi predstavljanja znanja prema različitim vrstama problema naglašavaju jednu vrstu informacija o problemu, a drugu ignorišu.

Znanje se može klasifikovati kao: **deklarativno (faktičko) znanje, proceduralno znanje i podrazumijevano znanje** [1].

Deklarativno znanje se definiše kao znanje da je nešto istinito ili lažno. To su činjenice, znanja koja opisuju neki element u posmatranoj oblasti (statičko stanje). Dakle, ne govori se ništa o dinamičkim aktivnostima povezanim sa objektom, odnosno elementom u posmatranoj oblasti. Predstavlja se u obliku deklarativnih rečenica – Svijetlo na semaforu je crveno.

Proceduralno znanje se obično definiše kao znanje o tome kako učiniti nešto (pravila, strategije, agende, procedure). Takođe se može reći da proceduralno znanje opisuje neke dinamičke akcije povezane sa elementima domena – činjenicama koje važe u trenutnom stanju svijeta. Primjer proceduralnog znanja jeste znanje o tome kako izvršiti sterilizaciju instrumenata u stomatološkoj ordinaciji, šta učiniti u slučaju da ekspertni sistem za automatsko upravljanje vozilom naiđe na upaljeno crveno svijetlo na semaforu i slično. Opšti oblik u kojem se predstavlja proceduralno znanje su zapravo pravila tipa:

```
AKO (skup uslova)

TADA (akcije koje treba preduzeti ili nove činjenice koje treba ubaciti u bazu znanja).

AKO Svijetlo na semaforu je crveno TADA zaustavi vozilo.
```

Uspješan eskpertni sistem mora biti zasnovan na kombinaciji deklarativnih i proceduralnih znanja.

Podrazumijevano znanje je ono što nazivamo nesvjesnim znanjem, jer ne može biti izraženo riječima. Primjer podrazumijevanog znanja jeste znanje o tome kako pomjeramo ruku, hodamo, vozimo bicikl.

Iznad znanja je **metaznanje**. Metaznanje je znanje o znanju i ekspertizi. To je znanje o tome kako upotrijebiti i upravljati znanjem. Veoma je bitno u ekspertnim sistemima. Naime, ekspertni sistemi mogu biti dizajnirani tako da posjeduju znanje o više različitih oblasti i da su ta znanja smještena u više različitih baza znanja. Metaznanje specificira koja je baza znanja odgovarajuća za rješavanje konkretnog problema. Na primjer, neka jedan ekspertni sistem posjeduje baze znanja o načinu na koji se vrši opravka auta tipa Renault, Peugeout i Fiat. Zavisno od toga koje je auto neophodno opraviti, različite baze znanja će se koristiti. Naime, bilo bi neefikasno sa stanovišta radne memorije i brzine da se sve baze koriste odjednom. Osim toga, neminovno bi došlo do konflikata ukoliko bi ekspertni sistem pokušao da primijeni pravila iz sve tri baze odjednom. Metaznanje se takođe može upotrijebiti i u okviru jednog domena znanja za odlučivanje koja grupa pravila je odgovarajuća u konkretnom slučaju.

3.2 Formalna (matematička) logika

Znanje može biti predstavljeno simbolima logike, pri čemu logika određuje postupke ispravnog rasuđivanja (analogno aritmetici). Bitan dio rasuđivanja je izvođenje zaključaka iz premisa.

Baza znanja je sastavljena od rečenica koje mogu predstavljati i parcijalno znanje. Ove rečenice su izražene u skladu sa sintaksom odabranog jezika predstavljanja znanja. **Sintaksa** specificira pravila koja moraju biti ispoštovana da bi se neka rečenica smatrala ispravno formiranom. Logika mora takođe definisati i semantiku jezika. Može se reći da se **semantika** bavi „značenjem" rečenice. Kada je u pitanju logika, definicija semantike je veoma precizna. Semantika jezika definiše istinitost svake rečenice u zavisnosti od svakog mogućeg stanja svijeta. U formalnoj logici, svaka rečenica mora biti istinita ili lažna u svakom mogućem stanju svijetu. Ne može biti nešto između istinitog i lažnog. Često se umjesto termina moguće stanje svijeta, koristi termin **model**. Pod modelom se podrazumijeva matematička apstrakcija kojom se svakoj bitnoj rečenici pridružuje stanje **istinita** ili **neistinita**.

Najraniji oblik formalne logike je razvio grčki filosof Aristotel u četvrtom vijeku prije nove ere. Aristotelova logika je zasnovana na silogizmima. Aristotel je razvio četrnaest silogizama, a još pet je razvijeno u srednjem vijeku. Silogizmi se sastoje od dvije **premise** i jednog **zaključka**, koji se izvodi iz premisa.

Klasičan primjer silogizma je:

Premisa 1: Svaki čovjek je smrtan.

Premisa 2: Sokrat je čovjek.

Zaključak 3: Sokrat je smrtan.

Ako su premise istinite, "logički slijedi" zaključak. Dakle, premise obezbjeđuju dokaze iz kojih zaključak neminovno slijedi. Prethodni silogizam se može predstaviti u opštoj formi:

1. Svaki α ima obilježje β.
2. γ je α.
3. γ ima obilježje β.

Za ovakav način zaključivanja je karakteristično da je nezavisno od konteksta, za razliku od prirodnih jezika gdje postoji zavisnost od konteksta. "Logički slijedi" samo na osnovu oblika (forme), a ne na osnovu sadržaja (konteksta). Otuda i naziv formalna logika. Može se reći da matematička ili formalna logika daje mehanizam zaključivanja u kojem je "logički izveden" zaključak barem tako dobar kao polazne pretpostavke. Sa druge strane, niti jedan formalan mehanizam ne može osigurati istinite polazne pretpostavke. Može se zaključiti da je navedeni mehanizam formalne logike dosljedan (iz istinitih premisa se izvode istiniti zaključci).

3.3. Propoziciona logika

U propozicionoj logici se koristi **deklarativan način predstavljanja znanja**, odnosno akumulacija statičkih činjenica sa ograničenim informacijama o tome kako ih koristiti. Propoziciona logika pretpostavlja da se svijet sastoji od činjenica. Propoziciona logika je nezavisna od konteksta. Njom se deklarativne rečenice (koje mogu biti istinite ili lažne) preslikavaju u sistem simbola. Npr.: "Uran je radioaktivni hemijski element", preslikava se u simbol P. Nedostatak propozicione logike je činjenica da je veoma izražajno limitirana (za razliku od prirodnih jezika) [7].

Da bi se shvatio način zaključivanja propozicione logike neophodno je detaljno se upoznati sa njenom semantikom i sintaksom. Takođe je potrebno upoznati mehanizam nalijeđivanja/prenosivosti – relacije između neke rečenice i druge rečenice koja iz nje slijedi i utvrditi kako ovo vodi do jednostavnog algoritma za logičko zaključivanje.

3.3.1 Sintaksa propozicione logike

Propoziciona logika se može nazvati i iskaznom logikom. Sintaksa propozicione logike definiše dozvoljene rečenice. **Atomska rečenica** je nedjeljivi sintaksni element – sastoji se od jednog iskaznog simbola. Svaki **iskazni simbol** predstavlja iskaz koji može biti tačan ili netačan. Za obilježavanje iskaznih simbola se koriste velika slova pri kraju engleskog alfabeta (P, Q, R, S, T). Imena iskaznih simbola su proizvoljna, ali se najčešće daju tako da imaju neko mnemoničko značenje za korisnika.

Postoje dva iskazna simbola sa fiksnim značenjem (konstante): **true** (istinito, $\top$) je uvijek istinit iskaz, i **false** (neistinitost, $\perp$) koji je uvijek neistinit iskaz. Ovi iskazni simboli se nazivaju **simbolima istinitosti.**

Za formiranje **složenih iskaza – formula** koriste se **simboli povezivanja**, odnosno, logički operatori koji povezuju jednostavne iskaze, da bi formirali složene iskaze.

Postoji pet simbola povezivanja u uobičajenoj upotrebi:

1. $\neg$ (ne, not, negacija, ~) – veznik kojim se mijenja istinitost iskaza, true postaje false i obratno. Iskaz oblika $\neg P$ se naziva negacija *P*-a. Sada se može uvesti i pojam literala kao atomskog iskaza (pozitivan literal) ili negacije atomskog iskaza (negativan literal).
2. $\wedge$ (i, and, konjunkcija) – koristi se za indikaciju da svaki iskaz u složenom iskazu formiranom povezivanjem iskaza korišcenjem ovog veznika mora biti istinit, da bi dobijeni iskaz bio istinit. Cijela se formula naziva konjunkcija, a svaka komponenta konjunkt.
3. $\vee$ (ili, or, disjunkcija) – koristi se za indikaciju da će složeni iskaz formiran povezivanjem iskaza korišćenjem ovog veznika biti istinit, ukoliko je bilo koji od ovako povezanih iskaza (ili svi) istinit. Svaka komponenta se naziva disjunkt, a cio iskaz disjunkcija.
4. $\Rightarrow$ (ako, if, $\rightarrow$, imlikacija) – koristi se za pravljenje ako-tada konstrukcija. Primjer bi bio iskaz oblika P $\Rightarrow$ Q. Cio iskaz se naziva implikacija (ili kondicional, pravilo), P je premisa, Q je zaključak ili posljedica.
5. $\Leftrightarrow$ (akko, iff, $\leftrightarrow$, $\equiv$, ekvivalencija) – označava logičku ekvivalenciju dvije formule. P $\Leftrightarrow$ Q označava da su stanja istinitosti lijeve i desne strane ekvivalentna. Često se naziva i bikondicionalom.

Formalna gramatika propozicione logike se može predstaviti na sledeći način:

1. Svaki iskazni simbol i simbol istinitosti je rečenica (atomska rečenica) – true, P, Q, i R su četiri rečenice.
2. Ako su P i Q rečenice, onda su rečenice: ($\neg$P), ($\neg$Q), (P$\wedge$Q), (P$\vee$Q), (P$\Rightarrow$Q), (P$\Leftrightarrow$Q).

Dakle, negacija rečenice je rečenica ($\neg$ P i $\neg$ false su rečenice), konjunkcija ($\wedge$) dvije rečenice je rečenica (P $\wedge$ $\neg$ Q je rečenica), disjunkcija ($\vee$) dvije rečenice je rečenica (P $\vee$ $\neg$ Q je rečenica), implikacija jedne rečenice u drugu je rečenica (P $\Rightarrow$ Q je rečenica), ekvivalencija dvije rečenice je rečenica. Iskazne rečenice se grade samo konačnom primjenom prethodnih pravila. Legalne rečenice se nazivaju i **dobro formiranim formulama ili DFF**.

U rečenicama iskaznog računa simboli zagrada (,),[,], se koriste da bi se grupisali simboli u podiskaze i tako kontrolisao redoslijed izračunavanja. Postoje tumačenja po kojima bi trebalo svaki iskaz koji je dobijen korišćenjem binarnog operatora pisati u okviru zagrada [2]. Na ovaj način se obezbjeđuje kompletna nedvosmislenost sintakse. To takođe znači da se mora pisati ((A $\wedge$ B) $\Rightarrow$ C) umjesto A $\wedge$ B $\Rightarrow$ C. Međutim, u cilju poboljšavanja preglednosti često se izostavljaju zagrade, pri čemu se oslanja na prioritet veznika (operatora). Prioriteta simbola povezivanja u propozicionoj logici je (od najvećeg ka najmanjem): $\neg$, $\wedge$, $\vee$, $\Rightarrow$, $\Leftrightarrow$.

Stoga je rečenica $\neg$ P $\vee$ Q $\wedge$ R $\Rightarrow$ S ekvivalentna rečenici (($\neg$ P) $\vee$ (Q $\wedge$ R)) $\Rightarrow$ S. Prioritet ne rješava dvosmislenost u slučaju iskaza oblika A $\wedge$ B $\wedge$ C, koji se mogu čitati kao ((A $\wedge$ B) $\wedge$ C) ili kao (A $\wedge$ (B $\wedge$ C)). Semantika određuje značenje različitih očitavanja. Prethodna dva čitanja daju isti rezultat i dozvoljena su. S druge strane rečenice oblika A $\Rightarrow$ B $\Rightarrow$ C nisu dozvoljene jer dva moguća različita očitavanja imaju različita značenja. U ovakvim slučajevima je obavezno korišćenje zagrada.

3.3.2 Semantika propozicione logike

Semantika definiše pravila za određivanje istinitosti iskaza/rečenice u odnosu na određeni model. U propozicionoj logici, model fiksira vrijednost istinitosti (istinit, nije istinit) za svaki propozicioni simbol. Na primjer, ukoliko neki iskaz u bazi znanja koristi propozicione simbole P, Q i R, jedan mogući model je:

m_1 = *{P=false, Q=false, R= true).*

Jasno je da sa tri propoziciona simbola postoji $2^3 = 8$ mogućih modela. Analogno, ukoliko bismo imali *n* propozicionih simbola, postojalo bi 2^n mogućih modela. Takođe se može primijetiti da ovakvim načinom modeliranja znanja, modeli postaju čisto matematički objekti, bez ikakve konekcije sa konkretnim problemom. *P* može biti „Danas je lijep dan“ ili „Ja sam u Parizu danas i sjutra“.

Semantika propozicione logike mora specificirati i na koji način se računa istinitost bilo kojeg iskaza, za dati model. U propozicionoj logici, ovo se čini **rekurzivno**, polazi se od činjenice da su sve rečenice sastavljene iz atomskih rečenica i pet veznika. Stoga je neophodno definisati na koji način se računa istinitost atomskih rečenica i na koji način se računa istinitost rečenica formiranih pomoću svakog od ovih pet veznika.

Istinitost atomskih rečenica se jednostavno određuje:

- true je istinito i false lažno-neistinito u svakom modelu;
- Istinitost svakog drugog propozicionog simbola mora biti specificirana u samom modelu za koji se računa istinitost nekog iskaza. Na primjer u ranije datim modelu P je neistinito (false).

Za kompleksne rečenice postoje pravila koja svode istinitost kompleksnih rečenica na istinitost jednostavnijih rečenica. Pravilo za svaki operator povezivanja može biti dato na pregledan način **tabelom istinitosti**, koja specificira istinitost kompleksnog iskaza za svaku moguću kombinaciju istinitosti njegovih komponentni (svaki model). Tabela istinitosti za pet navedenih operatora povezivanja je data Tabelom 1, gdje je su sa $\top$ i $\perp$ označeni simboli istinitosti (istinito i neistinito, respektivno).

Korišćenjem ovakvih tabela, može se izračunati istinitost bilo kojeg iskaza *S* u odnosu na bilo koji model *m* korišćenjem jednostavne procedure rekurzivne procjene. Na primjer, iskaz $\neg P \wedge (Q \vee R)$ procijenjen u m_1 rezultuje u *true* $\wedge$ *(false* $\vee$ *true)* = *true* $\wedge$ *true* = *true*.

Ranije je rečeno da je baza znanja sastavljena od skupa rečenica. Sada se može reći da baza znanja zapravo predstavlja konjunkciju ovih rečenica. Odnosno, ukoliko se startuje sa praznom bazom znanja, i u nju ubace iskazi S_1, S_2, ..., S_n, može se reći da je BZ= $S_1 \wedge S_2 \wedge ... S_n$. Ovo znači da se baza znanja može tretirati kao naizmjenične rečenice. Dodjeljivanje istinitosne vrijednosti iskaznoj rečenici se naziva **interpretacijom**, a to je tvrdnja o istinitosti te rečenice u nekom mogućem modelu/stanju svijeta – bazi znanja.

P	Q	$\neg P$	$P \wedge Q$	$P \vee Q$	$P \Rightarrow Q$	$P \Leftrightarrow Q$
$\perp$	$\perp$	T	$\perp$	$\perp$	T	T
$\perp$	T	T	$\perp$	T	T	$\perp$
T	$\perp$	$\perp$	$\perp$	T	$\perp$	$\perp$
T	T	$\perp$	T	T	T	T

Tabela 1 Tabela istinitosti za logičke veznike

Ako se pogleda Tabela 1, može se zaključiti da je vrijednost istinitosti za negaciju, konjunkciju i disjunkciju u saglasnosti sa onim što intuitivno podrazumijevamo pod značenjem riječi „ne“, „i“, „ili“. Tabela istinitosti za implikaciju $\Rightarrow$ može djelovati nejasno na prvi pogled. Naime, može se desiti da ne odgovara u potpunosti nečijem intuitivnom shvatanju značenja „P uzrokuje Q“ ili „ako P, onda Q“. Implikacija u propozicionoj logici je materijalna implikacija. Namjera materijalne implikacije je modelirati uslovnu konstrukciju, (a ne uzročno-posljedičnu vezu), t.j.: ako P tada Q, t.j. ako je P istinit, tada je (P $\Rightarrow$ Q) istinito samo ako je Q istinito.

Neintuitivni aspekt materijalne implikacije se može pokazati kroz rečenicu oblika (1 + 1 = 2) $\Rightarrow$ (“Podgorica je glavni grad Crne Gore”). Ova rečenica je istinita formula u propozicionoj logici, jer su prethodna (premisa) (P) i posljedična (zaključna) (Q) tvrdnja istinite. Sa druge strane, ovakva rečenica se bez sumnje smatra čudnom kada je u pitanju prirodan jezik.

Drugi razlog za zabunu vezanu za tabelu istinitosti implikacije jeste činjenica da je implikacija istinita, uvijek kada je premisa neistinita. Tako bi obje rečenice (1+ 1 = 3) $\Rightarrow$ (“Podgorica je glavni grad Crne Gore”) i (1+ 1 = 3) $\Rightarrow$ (“Pariz je glavni grad Crne Gore”) bile istinite. Ovo se može logički tumačiti ukoliko se (P $\Rightarrow$ Q) posmatra na sledeći način: Ukoliko je premisa P istinita, tada možemo tvrditi da je cijela implikacija istinita. U suprotnom, ne možemo ništa tvrditi. Naime, u prirodnom jeziku mogli bi ovakvim formulama implikacije dodijeliti bilo istinitost ili neistinitost, a možda čak i tvrditi da ako je prethodna tvrdnja (P) neistinita, implikacija ne mora biti ni istinita ni neistinita. Međutim, u formalnoj logici prihvaćena je konvencija: **Ako je P neistinit, tada je implikacija (P $\Rightarrow$ Q) istinita, nezavisno o istinitosti Q.** Jedini način da cijela implikacija bude neistinita jeste ukoliko je premisa P istinita a posljedična tvrdnja Q neistinita. Tako bi (1+ 1 = 2) $\Rightarrow$ (“Pariz je glavni grad Crne Gore”) bila neistinita tvrdnja.

Na pitanje zašto ima smisla ovakva tabela istinitosti za implikaciju može se odgovoriti kroz sledeće rezonovanje [5]: Koje su moguće opcije?

- Za P = istinito jasna je istinitost implikacije (istinita ili neistinita zavisno o Q).
- Za P = neistinito postoje 4 moguće tablice:

Ako se prihvati tablica 1, to je konjunkcija, ako se prihvati tablica 2, to je Q, ako se prihvati tablica 3, to je ekvivalencija. Dakle, preostaje jedino četvrta tablica.

P	Q	1	2	3	4
⊥	⊥	⊥	⊥	T	T
⊥	T	⊥	T	⊥	T
T	⊥	⊥	⊥	⊥	⊥
T	T	T	T	T	T

Tabela istinitosti za ekvivalenciju P $\Leftrightarrow$ Q pokazuje da je ekvivalencija istinita kad god su oba iskaza P $\Rightarrow$ Q i Q $\Rightarrow$ P istinita. U našem jeziku ovo se često piše kao „P ako i samo ako Q“ ili skraćeno „P akko Q“. Može se reći da su dvije formule ekvivalentne ili jednake ako imaju jednaku (istu) istinitosnu vrijednost za svaku interpretaciju.

3. 3. 2 Zaključivanje u propozicionoj logici

Cilj logičkog zaključivanja u ekspertnim sistemima je utvrđivanje istinitosne vrijednost iskaza *a* u modelu koji čini baza znanja (BZ) (istinitosne vrijednosti iskaza smještenih u bazi znanja). Svi iskazi u BZ se smatraju istinitim, pri čemu neki od njih mogu biti prikazani kao negacije. Utvrđivanje istinitosne vrijednosti iskaza u modelu koji čini BZ uključuje relaciju logičke implikacije ili povlačenja (engl. entailment) među iskazima (rečenicama), npr. iskaz *b* je logička posljedica iskaza *a*, što se uobičajeno matematički označava kao: ***a*|=*b* (*b* je logička posljedica iskaza *a*)**. Formalna definicija logičke implikacije bi bila *a*|=*b* akko je u svakom modelu u kojem je *a* istinito, i *b* takođe istinito. Neformalno se može reći da je istinitost iskaza *b* „sadržana“ u istinitosti iskaza a.

Ukoliko se algoritmom zaključivanja *i* iz BZ može izvesti *a*, piše se: ***BZ*|-$_i$*a***, a čita se „*a* je izvedeno iz BZ korišćenjem algoritma zaključivanja *i*“. Za algoritam zaključivanja kojim se izvode samo iskazi koji su logičke posljedice polaznih iskaza se kaže da je **dosljedan** ili **onaj koji zadržava istinitost**. Dosljednost je jako poželjna osobina algoritama zaključivanja. Može se reći da je algoritam zaključivanja *i* dosljedan ako ***BZ*|=*a*** kad god ***BZ*|-$_i$*a***, t.j. svaki iskaz dokazan algoritmom *i* na modelu BZ je ujedno i logička posljedica skupa iskaza sadržanih u BZ. **Kompletnost** je takođe poželjna osobina algoritama za zaključivanje. Algoritam za zaključivanje je kompletan ukoliko se njime može izvesti bilo koja rečenica koja je posledica premisa modela nad kojim se vrši zaključivanje. Može se pisati ***BZ*|-$_i$*a*** kad god ***BZ*|=*a***, t.j. svaku logičku posljedicu skupa BZ moguće je dokazati pravilima *i* – utvrditi njenu istinitost.

Formalna logika je dosljedna, kompletna i odrediva (može se interpretirati svaki iskaz, npr. preslikavanjem u tabelu istinitosti), jer operiše s konačnim skupom simbola te samim tim ima konačan skup modela za ispitivanje.

Validnost, ekvivalencija i zadovoljivost propozicione logike

Poput logičke implikacije, ekvivalencija, validnost i zadovoljivost su koncepti primijenljivi na sve oblike logike, ali se najbolje mogu ilustrovati kroz konkretnu logiku, u našem slučaju propozicionu logiku.

Koncept logičke validnosti/valjanosti – iskaz je validan ako je istinit u svim modelima. Na primjer, $A \vee \neg A$ je validan iskaz. Validni iskazi su takođe poznati i kao **tautologije**.

Ukoliko se ima na umu definicija logičkog zaključivanja, može se izvesti definicija za dedukciju teorema (**dokazivanje teorema dedukcijom**) [2]:

Za bilo koje iskaze a i b, važi $a|=b$ *akko je iskaz* $(a \Rightarrow b)$ *validan (tautologija).*

Dakle, ne postoji model za koji bi iskaz $(a \Rightarrow b)$ bio neistinit (što bi se moglo desiti samo u slučaju da je iskaz *a* istinit a iskaz *b* neistinit, Tabela 1)

Tautologije su vrsta formula koje imaju veliku primjenu [3]. Tautologije se mogu direktno koristiti za zaključivanje u propozicionoj logici [4]. To su formule koje su istinite na osnovu svoje forme, odnosno načina na koje su izgrađene korišćenjem simbola logičkih veznika, a ne na osnovu neke posebne vrijednosti iskaznih simbola od kojih su sastavljene. Tautologije su zato obrasci ispravnog zaključivanja. Pošto je negacija svake tautologije kontradikcija – uvijek neistinita, ove formule se često koriste kao pokazatelji stranputica do kojih se došlo u zaključivanju.

Formula $((P \Rightarrow Q) \wedge P) \Rightarrow Q$ je jedna tautologija. Već je rečeno da se ovo neposredno provjerava ispisivanjem sve četiri različite interpretacije iskaznih simbola P i Q, računanjem istinitosnih vrijednosti podformula formule i, konačno, računanjem istinitosne vrijednosti same formule koja se razmatra:

P	Q	$P \Rightarrow Q$	$(P \Rightarrow Q) \wedge P$	$((P \Rightarrow Q) \wedge P) \Rightarrow Q$
T	T	T	T	T
T	⊥	⊥	⊥	T
⊥	T	T	⊥	T
⊥	⊥	T	⊥	T

Sličnim razmatranjem se ustanovljava da je formula $(P \Rightarrow Q) \wedge (P \wedge \neg Q)$ kontradikcija.

Neke od važnih tautologija su (sa n su označene uvek istinite formule, dok o označava neistinite formule):

1. $\neg \neg A \Rightarrow A$ (zakon dvojne negacije),
2. $(A \wedge B) \Leftrightarrow (B \wedge A)$ (zakon komutativnosti za $\wedge$),
3. $(A \vee B) \Leftrightarrow (B \vee A)$ (zakon komutativnosti za $\vee$),
4. $((A \wedge B) \wedge C) \Leftrightarrow (A \wedge (B \wedge C))$ (zakon asocijacijativnosti za $\wedge$),
5. $((A \vee B) \vee C) \Leftrightarrow (A \vee (B \vee C))$ (zakon asocijacijativnosti za $\vee$),
6. $(A \wedge (B \vee C)) \Leftrightarrow ((A \wedge B) \vee (A \wedge C))$ (zakon distribucije $\wedge$ prema $\vee$),
7. $(A \vee (B \wedge C)) \Leftrightarrow ((A \vee B) \wedge (A \vee C))$ (zakon distribucije $\vee$ prema $\wedge$),
8. $(A \Rightarrow B) \Leftrightarrow (\neg A \Rightarrow \neg B)$ (kontrapozicija),

9. $(A \Rightarrow B) \Leftrightarrow (\neg A \vee B)$ (zakon uklanjanja implikacije),
10. $(A \Leftrightarrow B) \Leftrightarrow ((A \Rightarrow B) \wedge (B \Rightarrow A))$ (zakon uklanjanja ekvivalencije),
11. $(\neg(A \wedge B)) \Leftrightarrow (\neg A \vee \neg B)$ (zakon De Morgana),
12. $(\neg(A \vee B)) \Leftrightarrow (\neg A \wedge \neg B)$ (zakon De Morgana),
13. $(A \wedge A) \Leftrightarrow A$ (zakon idempotencije za $\wedge$ - jednaka važnost),
14. $(A \vee A) \Leftrightarrow A$ (zakon idempotencije za $\vee$ - jednaka važnost).
15. $A \Rightarrow A$ (zakon refleksivnosti implikacije),
16. $A \vee \neg A$ (zakon isključenja trećeg),
17. $\neg (A \wedge \neg A)$ (zakon neprotivrječnosti),
18. $((A \Rightarrow B) \wedge (B \Rightarrow C)) \Rightarrow (A \Rightarrow C)$ (zakon tranzitivnosti za implikaciju),
19. $((A \Leftrightarrow B) \wedge (B \Leftrightarrow C)) \Rightarrow (A \Leftrightarrow C)$ (zakon tranzitivnosti za ekvivalenciju),
20. $(\neg A \Rightarrow (B \wedge \neg B)) \Rightarrow A$ (zakon svođenja na apsurd),
21. $(A \wedge (A \vee B)) \Leftrightarrow A$ (zakon apsorpcije),
22. $(A \vee (A \wedge B)) \Leftrightarrow A$ (zakon apsorpcije),
23. $(A \wedge (A \Rightarrow B)) \Rightarrow B$ (modus ponens),
24. $((A \Rightarrow B) \wedge \neg B) \Rightarrow \neg A$ (modus tolens),
25. $(A \vee n) \Leftrightarrow n$ (zakon disjunkcije sa tautologijom),
26. $(A \wedge n) \Leftrightarrow A$ (zakon konjunkcije sa tautologijom),
27. $(A \vee o) \Leftrightarrow A$ (zakon disjunkcije sa kontradikcijom),
28. $(A \wedge o) \Leftrightarrow o$ (zakon konjunkcije sa kontradikcijom).

Za ove formule, metodom istinitosnih tabela, se može utvrditi da jesu tautologije.

Koncept logičke ekvivalencije – Dva iskaza su logički ekvivalentna ili jednaka ako su istinita u istom skupu modela. Matematički zapis je $a \equiv b$. Na primjer, jednostavno se može dokazati, koristeći tabele istinitosti da su iskazi $(A \wedge B)$ i $(B \wedge A)$ logički ekvivalentni. Lako je zaključiti da su sve tautologije logički ekvivalentni iskazi.

Ekvivalencija se može definisati i u svijetu logičkih posljedica kao:

$$a \equiv b \textit{ akko } (a \models b) \quad i \quad (b \models a).$$

Semantička ekvivalencija je na taj način identična dokazivoj ekvivalenciji (t.j. ako želiš dokazati ekvivalentnost, dokaži da je $((a \Rightarrow b) \wedge (b \Rightarrow a))$ istinit za svaki mogući model – tautologija), [5].

Koncept zadovoljivosti (engl. satisfiability) – Iskaz je zadovoljiv ukoliko je istinit u nekom (barem jednom) modelu. Princip zadovoljivosti vrijedi i za skup iskaza. U skladu sa ovom definicijom može se reći da je iskaz nezadovoljiv ukoliko ne postoji model u kojem je istinit. Ukoliko je iskaz a istinit u modelu m, kažemo da je m model iskaza a ili da m zadovoljava iskaz a. Zadovoljivost se može utvrditi tako što se navode mogući modeli dok se ne nađe jedan koji zadovoljava željeni iskaz.

Stoga se ovakav način dokazivanja često naziva i **pravilom nabrajanja modela**. Ukoliko *a* sadrži *n* propozicionih simbola, postoji 2^n mogućih modela. Stoga je kompleksnost algoritma dokazivanja zadovoljivosti pretraživanjem mogućih modela reda $O(2^n)$ [2].

Mnogi problemi u kompjuterskim naukama su u stvari zadovoljivi problemi. Odgovarajućim transformacijama se ispitivani problem može riješiti (naći model za koji je istinit ili dokazati da nije istinit ni u jednom modelu) tako što se provjerava da li je zadovoljiv.

Validnost i zadovoljivost su povezani: *a* je validna akko je $\neg a$ nezadovoljiva. Važi i obratno, *a* je zadovoljiva akko $\neg a$ nije validna. Takođe važi i:

$$a \models b \text{ akko su iskazi } (a \wedge \neg b) \text{ nezadovoljivi.}$$

Dokazivanje iskaza *b* na osnovu iskaza *a* provjeravanjem nezadovoljivosti iskaza $(a \wedge \neg b)$ odgovara tehnici dokazivanja teorema koja je u standardnoj matematici poznata kao *reductio ad absurdum* (redukcija do apsurda) [2]. Ovaj način dokazivanja se naziva i **dokazivanje kontradikcijom** – opovrgavanjem. Naime, pretpostavlja se da je iskaz *b* neistinit i pokazuje se da ova pretpostavka dovodi do kontradikcije sa aksiomom *a* (iskazom za koji se zna da je istinit). Ova kontradikcija je upravo ono što se misli kada se kaže da je iskaz $(a \wedge \neg b)$ nezadovoljiv.

Istinitosne tabele se koriste i za utvrđivanje da li se neka formula može zadovoljiti ili je kontradikcija, tj. ne može se zadovoljiti.

Šabloni rezonovanja u formalnoj logici

Postoji nekoliko standardnih šablona rezonovanja koji se mogu primijeniti kako bi se izveo niz zaključaka koji bi vodili do željenog cilja – rješenja postavljenog problema. Ovi šabloni zaključivanja se nazivaju **pravila zaključivanja**.

Najpoznatije pravilo zaključivanja se naziva **Modus Ponens** i zapisuje se na sljedeći način [2]:

$$\frac{A \Rightarrow B, \qquad A}{B}$$

Uvedena oznaka znači, iskaz *B* može biti izveden kada god su, kao istiniti, dati iskaz oblika $A \Rightarrow B$ i iskaz *A*. Označimo sa *A*, na primjer, iskaz „pada kiša“ i sa *B* iskaz „zemlja je mokra“. Ako imamo iskaze $A \Rightarrow B$, odnosno „ako pada kiša onda zemlja je mokra“, i znamo da je iskaz A – „pada kiša“ istinit, korišćenjem Modus Ponensa zaključujemo da važi *B* – „zemlja je mokra“.

Drugo korisno pravilo zaključivanje je **pravilo uklanjanja konjunkcije**, koje kaže da se bilo koji od konjunkata može izvesti na osnovu konjunkcije:

$$\frac{A \wedge B}{A}$$

Za prethodni primjer bi značilo da ako znamo da je istinit iskaz $A \wedge B$ – „pada kiša i zemlja je mokra“, zaključujemo da je istinit i iskaz *A* – „pada kiša“. Analogno se može utvrditi i istinitost iskaza *B*.

Posmatranjem tabele istinitosti za *A* i *B*, može se pokazati da su i Modus Ponens i pravilo uklanjanja konjunkcije dosljedni algoritmi zaključivanja. Ovi algoritmi zaključivanja se stoga mogu smatrati šablonima zaključivanja koji se mogu primijeniti za bilo koju instancu (konkretno značenje iskaza *A*), generišući dosljedno zaključivanje – dokazivanje bez potrebe za nabrajanjem modela.

Svaka od navedenih tautologija se može upotrijebiti kao dosljedno pravilo zaključivanja. Na primjer, tautologija (10) $(A \Leftrightarrow B) \Leftrightarrow ((A \Rightarrow B) \wedge (B \Rightarrow A))$ (zakon uklanjanja ekvivalencije) se može iskoristiti za dobijanje dva pravila zaključivanja:

$$\frac{A \Leftrightarrow B}{(A \Rightarrow B) \wedge (B \Rightarrow A)} \qquad i \qquad \frac{(A \Rightarrow B) \wedge (B \Rightarrow A)}{A \Leftrightarrow B}$$

Postupak u kojem se sekvencionalno upotrijebljavaju date tautologije, odnosno pravila zaključivanja, kako bi se izveo željeni iskaz naziva se **dokaz zamjenom** [4]. Ovaj postupak je alternativa postupku u kojem se isprobavaju mogući modeli dok se ne dođe do nekog modela koji zadovoljava razmatrani iskaz. Dokazivanje se može vršiti tako što se polazi od iskaza iz BZ i ide naprijed kako bi se dokazao željeni iskaz (rezonovanje unaprijed), ili se može ići od ciljnog iskaza i pokušavati pronaći niz pravila zaključivanja (tautologije) koja bi dovela do početne BZ, odnosno nekog od iskaza koji se u njoj nalazi (rezonovanje unazad). Može se zaključiti da je složenost ovakvog načina dokazivanja u najgorem slučaju jednaka složenosti zaključivanja nabrajanjem modela. Međutim, u većini praktičnih slučajeva, pronalaženje dokaza može biti jako efikasno, iz prostog razloga što ignoriše sve nerelevantne iskaze iz BZ, bez obzira na to koliko ih ima. Dakle, posmatraju se samo oni iskazi koji se koriste tokom dokazivanja ili koji su uključeni u toku dokazivanja.

Dodavanjem novih iskaza u BZ ne povećava se složenost algoritma ukoliko neki od dodatih iskaza nije povezan sa iskazom koji se dokazuje. Ovo svojstvo logičkih sistema slijedi iz fundamentalnog svojstva zvanog **monotonost**. Monotonost tvrdi da se skup rečenica dobijenih logičkom implikacijom može samo povećavati dodavanjem novih iskaza (informacija) u bazu znanja. Dakle, za iskaze *A* i *B* važi:

$$ako\ KB|{=}A\ onda\ KB \wedge B|{=}A.$$

Na primjer, pretpostavimo se baza znanja na osnovu koje smo donijeli zaključak da je iskaz „zemlja je mokra“ istinit, pored iskaza „pada kiša“ i „ako pada kiša onda zemlja je mokra“, doda iskaz „ako pada kiša onda poželjno je ponijeti kišobran“. Dodati iskaz ne utiče na donijeti zaključak. Dodatni iskaz može pomoći da se izvuku neki novi zaključci, ali ne može učiniti nevaljanim već donijeti zaključak. Monotonost dakle znači da se pravila zaključivanja mogu primijeniti kada god postoje odgovarajuće premise u bazi znanja – izvedeni zaključak mora logički slijediti iz BZ bez obzira na to šta se još nalazi u bazi. U slučaju prostog nabrajanja modela, složenost bi eksponencijalno rasla dodavanjem novih iskaza u BZ [2].

Rezolucija – razriješavanje

Za pravila zaključivanja navedena u prethodnom izlaganju je rečeno da su dosljedna. Međutim, nije rečeno ništa o kompletnosti algoritama zaključivanja koji ih koriste. Ovi algoritmi nisu uvijek kompletni. Naime, ukoliko su dostupna pravila zaključivanja u datoj BZ neadekvatna za dostizanje postavljenog cilja, istinitost postavljenog cilja se ne može odrediti. Ovdje se pod neadekvatnim misli, da ne postoji mogućnost da se pravila iz BZ predstave u pogodnoj formi, iako se na osnovu njihovog značenja može doći do željenog cilja.

Rezolucija je pravilo zaključivanja koje je kompletno zaključivanje kada se kombinuje sa bilo kojim kompletnim algoritmom pretraživanja. **Puna rezolucija** se može izreći kao: logička posljedica dvaju istinitih, konjunkcijom vezanih klauzula je klauzula bez jednog komplementarnog para literala. **Klauzula** je formula oblika disjunkcije literala. Komplementarni par literala su nenegirani i negirani jednaki atomski simboli. Dakle:

$$\frac{A_1 \vee \cdots \vee A_k, \qquad m_1 \vee \cdots \vee m_n}{A_1 \vee \cdots \vee A_{i-1} \vee A_{i+1} \vee \cdots \vee A_k \vee m_1 \vee \cdots \vee m_{i-1} \vee m_{i+1} \vee \cdots \vee m_n}, \qquad (1.1)$$

gdje su svi iskazi *A* literali i A_i i m_i komplementarni literali (jedan je negacija drugog). Pravilo rezolucije uzima dvije klauzule i izvodi novu klauzulu.

U nekim slučajevima se može doći i do tzv. jedinične rezolucije:

$$\frac{A_1 \vee \cdots \vee A_k, \qquad m}{A_1 \vee \cdots \vee A_{i-1} \vee A_{i+1} \vee \cdots \vee A_k}$$

gdje su iskazi A_i i m komplementarni literali. Pravilo jedinične rezolucije uzima klauzulu i jedan literal i izvodi novu klauzulu.

Kada je u pitanju klauzula dužine 2, puna rezolucija se može pisati kao:

$$\frac{A_1 \vee A_2, \qquad \neg A_1 \vee A_3}{A_2 \vee A_3},$$

t.j. rezolucija uzima dvije klauzule i izvodi novu klauzulu koja sadrži sve literale, osim dva komplementarna literala iz originalne dvije klauzule.

Na osnovu definicije pravila rezolucije vidi se da je primjenljivo samo za disjunkciju literala. Moglo bi se pomisliti da je stoga bitno samo za primjenu u bazama znanja koji se sastoje od ovakvih disjunkcija. Međutim, pravilo rezolucije je kompletan algoritam zaključivanja za cijelu propozicionu logiku (bilo koji oblik iskaza) jer *je svaki iskaz u propozicionoj logici, logički ekvivalentan konjunkciji disjunkcija literala*. Za iskaz predstavljen u obliku konjunkcije disjunkcija literala se kaže da ima konjunktivnu normalnu formu (engl. conjunctive normal form – CNF).

Posmatrajmo proizvoljan iskaz

$$\neg(P \Rightarrow Q) \vee (R \Rightarrow P)$$

Pogledajmo na koji način se može prikazati u CNF:

1. Eliminiši implikaciju upotrebom ekvivalentnog "$\vee$" oblika (tautologije (9): $(A \Rightarrow B) \Leftrightarrow (\neg A \vee B)$ *(zakon uklanjanja implikacije)*):

$$\neg(\neg P \vee Q) \vee (\neg R \vee P)$$

2. Redukuj doseg negacije (pomak u desno) primjenom De Morgan-ovih pravila (tautologije (11) i (12)), te eliminiši dvostruke negacije (tautologija (1)):

$$(P \wedge \neg Q) \vee (\neg R \vee P)$$

3. Pretvori u CNF asocijativnim i distributivnim pravilima (4) – (7):

$$(P \vee \neg R \vee P) \wedge (\neg Q \vee \neg R \vee P),$$

te dalje:

$$(P \vee \neg R) \wedge (\neg Q \vee \neg R \vee P) = CNF\ oblik$$

Algoritam pune rezolucije se može predstaviti na sljedeći način:

Ulaz: Skup klauzula, koje se nazivaju **aksiomi** (sigurno su istinite) i cilj.

Izlaz: Test – da li je cilj moguće izvesti na osnovu aksioma.

Početak:

1. Formirati skup *S* od aksioma i negirananog iskaza koji predstavlja cilj. Tvrdimo da iz početnog skupa aksioma logički ne slijedi cilj.
2. Predstaviti svaki element iz S u CNF koristeći prethodno opisanu proceduru. Sada se skup S sastoji od konjunkcije (sve rečenice smatramo da su istinite)

$$[A_{11} \vee A_{12} \vee \ldots\ldots \vee A_{1n}] \wedge$$

$$[A_{21} \vee A_{22} \vee \ldots\ldots \vee A_{2n}] \wedge$$

$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

$$[A_{m1} \vee A_{m2} \ldots\ldots \vee A_{mn}]$$

3. Dok se ne dobije nulta klauzula (o) ili bude nemoguće vršiti dalje razrješavanje klauzula ponavljati:
   a) Odabrati bilo koje dvije klauzule iz S, takve da jedna sadrži negaciju literala a druga klauzula sadrži odgovarajući pozitivni literal (ne-negirani).
   b) Razriješiti ove dvije klauzule i nazvati rezultujući klauzulu **rezolvent**om (engl resolvent – razriješena klauzula), te je dodati početnom skupu S.
4. Ukoliko je dobijena nulta klauzula, odgovor je „cilj je dokazan“.

Teorema 1. *Pravilo rezolucije je dosljedno.*

Dokaz Dosljednost rezolucije se može dokazati na jednostavan način, posmatranjem literala A_i u (1.1). Ukoliko je A_i istinito, m_i je neistinito, te stoga $m_1 \vee \ldots \vee m_{i-1} \vee m_{i+1} \ldots \vee m_n$ mora biti istinito (početna pretpostavka je da je $m_1 \vee \ldots \vee m_n$ istinito), pa je i iskaz $A_1 \vee \ldots \vee A_{i-1} \vee A_{i+1} \ldots \vee A_k \vee m_1 \vee \ldots \vee m_{i-1} \vee m_{i+1} \ldots \vee m_n$ istinit. S druge strane, ako je A_i lažno, $A_1 \vee \ldots \vee A_{i-1} \vee A_{i+1} \ldots \vee A_k$ je istinito (početna pretpostavka je da je $A_1 \vee \ldots \vee A_k$ istinito), pa je i iskaz $A_1 \vee \ldots \vee A_{i-1} \vee A_{i+1} \ldots \vee A_k \vee m_1 \vee \ldots \vee m_{i-1} \vee m_{i+1} \ldots \vee m_n$ istinit. Dakle, bilo da je A_i istinito ili neistinito, izvedeni iskaz $A_1 \vee \ldots \vee A_{i-1} \vee A_{i+1} \ldots \vee A_k \vee m_1 \vee \ldots \vee m_{i-1} \vee m_{i+1} \ldots \vee m_n$ je istinit.

Za pravilo rezolucije se može reći da je kompletno. *Bilo koji kompletan algoritam pretraživanja, primijenjujući samo pravila rezolucije, može izvesti zaključke o logičkoj implikaciji među iskazima.* Postoji jedno upozorenje vezano za značenje kompletnosti. Naime, rezolucija je kompletna na specifičan način. Odnosno, ukoliko je dato da je A istinito, ne može se koristiti rezolucija za automatsko generisanje iskaza oblika $A \vee B$. Ipak, rezolucija se može koristiti da se dobije odgovor na pitanje da li je iskaz $A \vee B$ istinit. Stoga se ovakva vrsta kompletnosti naziva – kompletnost opovrgavanja, što znači da rezolucija uvijek može biti korišćena da potvrdi ili opovrgne neki iskaz [2].

Upotreba algoritma rezolucije za zaključivanje u propozicionoj logici se može ilustrovati kroz sljedeći primjer.

Primjer 1. Neka je data sljedeća baza znanja:

1. *Vlažnost-je-visoka $\vee$ nebo-je-oblačno.*
2. *Ukoliko nebo-je-oblačno onda padaće-kiša.*
3. *Ukoliko vlažnost-je-visoka onda vruće-je.*
4. *Nije-vruće.*

i neka je cilj: padaće-kiša.

Treba dokazati upotrebom teoreme rezolucije da je cilj moguće izvesti na osnovu date baze znanja. Svaka stavka u bazi znanja se smatra istinitom.

Dokaz: Prvo je neophodno predstaviti date klauzule u simboličkom obliku:

P=Vlažnost-je-visoka, Q= nebo-je-oblačno, R= padaće-kiša, S= vruće-je.

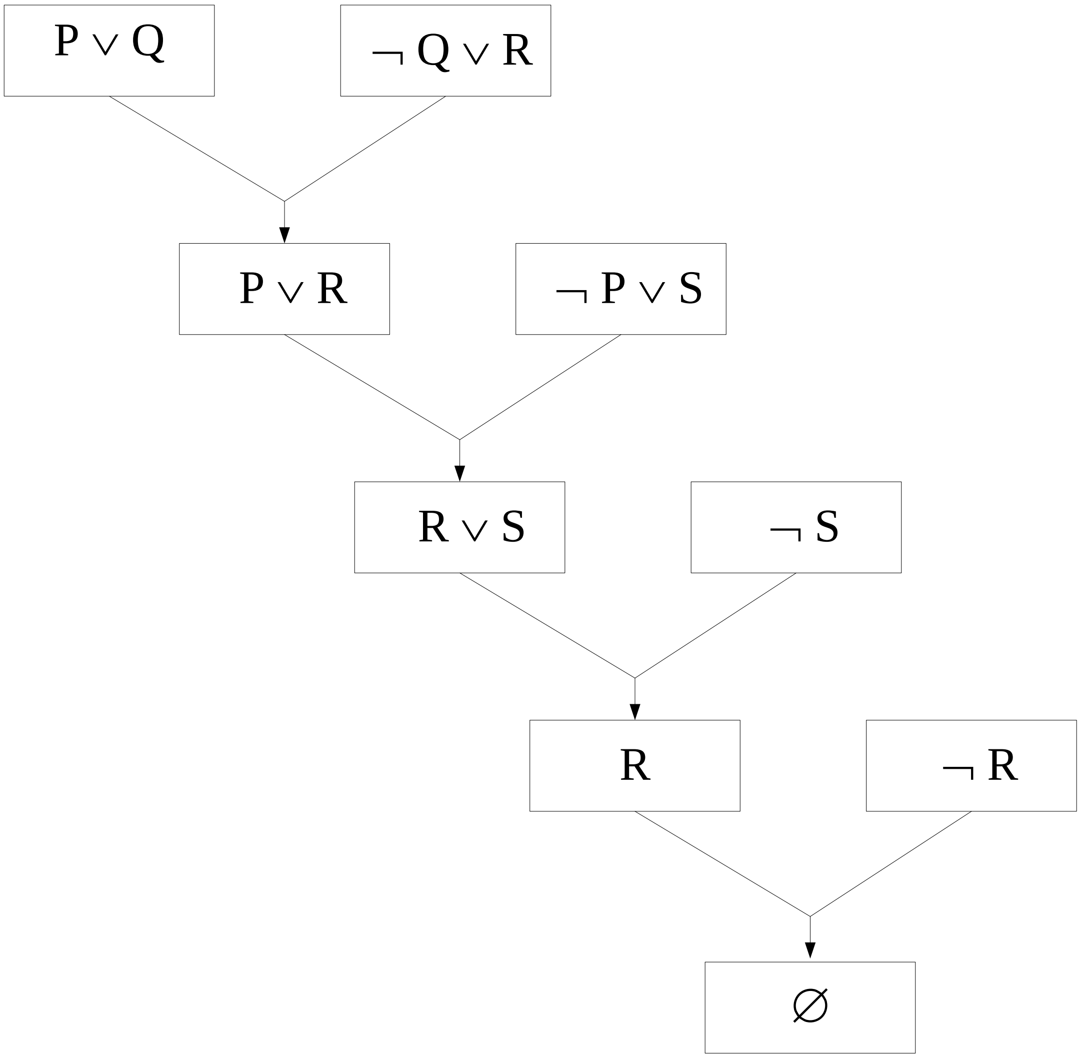
Baza podataka se sada može predstaviti kao:

1. $P \vee Q$
2. $Q \Rightarrow R$
3. $P \Rightarrow S$
4. $\neg S$

CNF forma gornjih klauzula postaje:

1. $P \vee Q$
2. $\neg Q \vee R$ *tautologija (9)*
3. $\neg P \vee S$ *tautologija (9)*
4. $\neg S$

Negirani cilj je $\neg R$. Sada treba postaviti skup S tako da sadrži svih 5 klauzula i primijeniti rezolutivni lagoritam. Rezolutivni algoritam je dat grafom na slici 1. Završava se nultom klauzulom, te se stoga zaključuje da je cilj dokazan i da će padati kiša.



Slika 1 Koraci algoritma rezolucije primijenjenog na rješavanje Primjera 1.

Primjer 2. Primenom rezolucije dokazati da je svaka od sledećih formula tautologija[1]*:*

a) $(P \Rightarrow Q) \Rightarrow [(R \vee P) \Rightarrow (R \vee Q)]$

b) $[(P \Rightarrow Q) \Rightarrow P] \Rightarrow P$

c) $(\neg P \Rightarrow P) \Rightarrow P$

d) $(P \Rightarrow Q) \Rightarrow (\neg Q \Rightarrow \neg P)$

Rješenje: U opštem slučaju dokazivanja neke propozicione formule rezolucijom, posjedujemo skup aksioma P i tvrđenje T. Rezoluciju primjenjujemo na skup $P \cup \neg T$ *to jest na skup koji se sastoji od polaznih aksioma i negacije tvrđenja. Ukoliko primjenom rezolucije dobijemo prazan stav, to jest protivriječnost u skupu* $P \cup \neg T$*, to znači da skup* $P \cup \neg T$ *ne može biti istinito ni za jedan model (semantiku koju dodjeljujemo pojedinim konstantama, promenljivima) aksioma i tvrđenja. Iz ovoga slijedi da je skup* $P \cup T$ *istinit u svakoj interpretaciji – za svaki model, shodno*

[1] Zadatak preuzet iz [6]

tome T je teorema koja se dobija iz polaznih aksioma P nezavisno od modela – tautologija.

Zamislimo sada da imamo samo tvrđenje T bez bilo kakvih aksioma. Ponovimo razmatranje iz prethodnog pasusa stavljajući da je P prazan skup: Ako rezolucijom pokažemo da ¬T nije istinito, dokazali smo da je T zadovoljeno nezavisno od interpretacije i nezavisno od ikakvih polaznih pretpostavki.

Propoziciona (iskazna) logika je podskup predikatske logike bez promenljivih. Formule iskazne logike koje su tačne za svaku interpretaciju (to jest, bez obzira koju istinitosnu vrednost imaju iskazi koje se pojavljuju u formuli) nazivaju se tautologije. Prema tome, da bismo dokazali da je neka formula tautologija, potrebno je primjenom rezolucije naći protivriječnost u skupu stavova koji predstavljaju negaciju polazne formule.

a) Negiramo datu formulu:

$\neg\{(P \Rightarrow Q) \Rightarrow [(R \vee P) \Rightarrow (R \vee Q)]\}$

Prevodimo dobijenu formulu u konjunktivnu normalnu formu (CNF).

Eliminacijom implikacije dobija se:

$$\neg\{\neg(\neg P \vee Q) \vee [\neg(R \vee P) \vee (R \vee Q)]\}$$

Negaciju spuštamo na atomske formule. Prebacimo najpre krajnje lijevu negaciju 'pod zagradu':

$$(\neg P \vee Q) \wedge (R \vee P) \wedge \neg(R \vee Q)$$

Transformisanjem posljednjeg člana formule dobijamo:

$$(\neg P \vee Q) \wedge (R \vee P) \wedge \neg R \wedge \neg Q$$

Podjelom na klauzule dobijamo traženu formu:

$$1.\ \neg P \vee Q$$

$$2.\ R \vee P$$

$$3.\ \neg R$$

$$4.\ \neg Q$$

Primjenom rezolucije tražimo protivrječnost u navedenim stavovima (stavovi za spajanje će biti birani tako da se cilj postigne u što manje koraka):

$$1.,\ 2. \rightarrow 5.\ Q \vee R$$

$$3.,\ 5. \rightarrow 6.\ Q$$

$$4.,\ 6. \rightarrow NIL$$

Dobijanjem protivrečnosti dokazano je da je polazna formula tautologija.

Minimum zahtijevanog znanja sa trećeg termina

1. Tipovi znanja?
2. Šta je meta znanje i zašto je bitno u ekspertnim sistemima?
3. Od čega se sastojala Aristotelova logika?
4. Šta je atomska rečenica u propozicionoj logici. Na koji način se obilježava. Koji su to iskazi sa fiksnim značenjem.
5. Navesti svih pet simbola povezivanja propozicione logike. Dati tabelu istinitosti za sve ove simbole.
6. Prokomentarisati tabelu istinitosti za logičku implikaciju.
7. Šta je dobro formirana rečenica u propozicionoj logici?
8. Koliko modela ima za iskaz $((\neg P) \vee (Q \wedge R)) \Rightarrow S$, navesti jedan (na kolokvijumu će se tražiti odgovor ovog tipa, za neki drugi iskaz. Navedeni iskaz služi za vježbu.).
9. Semantika propozicione logike – način na koji način se računa vrijednost istinitosti bilo kojeg iskaza, za dati model.
10. Šta znači $\boldsymbol{a} \models \boldsymbol{b}$, a šta $\boldsymbol{BZ} \vdash_{i} \boldsymbol{a}$?
11. Šta je dosljednost, a šta kompletnost algoritma? Zašto se ovo karakteristike zahtijevaju od algoritama koji se koriste u ekspertnim sistemima?
12. Koja je razlika između ekvivalencije i logične posljedičnosti iskaza u propozicionoj logici?
13. Kada se u propozicionoj logici za neki iskaz kaže da je zadovoljiv?
14. Šta je tautologija, a šta kontradikcija u propozicionoj logici?
15. Koja je mana Modus Ponens i pravila uklanjanja konjunknkcija? Koja im je prednost u odnosu na pravilo nabrajanja modela sa stanovišta kompleksnosti odgovarajućeg agoritma?
16. Svaki student će dobiti po jedan ili dva zadatka slična primjeru 1 ili primjeru 2.

LITERATURA

[1] Joseph C. Giarratano, Gary D. Riley.: *Expert Systems: Principles and Programming*, Prentice Hall, 2nd ed., 2002.

[2] Russell S., Norvig P.: *Artificial Intelligence: A Modern Approach*, Prentice Hall, NJ, 1995.

[3] Poliščuk, E.J.: *Ekspertni sistemi*, Informatička literatura JEP (vlastito izdanje), Podgorica, 2004.

[4] Konar A.: *Artificial Intelligence and Soft Computing: Behavioral and Cognitive Modeling of the Human Brain*, CRC Press, December 8, 1999.

[5] *http://www.zemris.fer.hr/predmeti/tes/nastava.html* - zadnji put pristupano, 10. 02. 2010. godine.

[6] Dragan Bojić, Dušan Velašević, Vojislav Mišić: *Zbirka zadataka iz Ekspertskih sistema*, Beograd, 1996.

[7] http://ri4es.etf.rs/, posljednji put pristupano, 18. 02. 2010. godine.