Automatsko rezonovanje – beleške sa predavanja Rezonovanje u logici prvog reda

Filip Marić
Milan Banković

*Matematički fakultet,
Univerzitet u Beogradu

Proletnji semestar 2018.

Pregled

Unutrašnja struktura iskaza

- Iskazna logika iskaze smatra elementarnim objektima i ne zalazi u njihovu internu strukturu.

Primer

- *Iskazi Sokrat je čovek. i Aristotel je čovek se smatraju različitim.*
- *Njihova untrašnja struktura je slična: . . . je čovek.*

- Poželjno je izgraditi bogatiju logiku koja bi analizirala i unutrašnju strukturu iskaza.

Odnosi među objektima

- Iskazi obično govore o svojstvima objekata i odnosima među objektima.

Primer

- *Sokrat je čovek* — $čovek(Sokrat)$.
- *Broj* 3 *je paran* — $paran(3)$.
- *Broj* 3 *je manji od broja* 5 — $manji(3, 5)$ *ili* $3 < 5$

- U zapisu iskaza, objekti su predstavljeni simbolima konstanti a svojstva i odnosi relacijskim simbolima.

Kombinovanje iskaza

- Iskaze je moguće kombinovati na isti način kao u iskaznoj logici.

Primer

- *Ako je broj* 4 *paran, onda je broj* 4 *veći od* 2 — $paran(4) \Rightarrow 4 > 2$.
- *Broj* 15 *je paran ili je deljiv brojem* 3 — $paran(15) \vee 3 \mid 15$.

Funkcije

- Moguće je govoriti o objektima koji su jednoznačno određeni na osnovu nekih datih objekata.

Primer

- *Sokratova majka je žena.* — *žena(majka(Sokrat)).*
- *Zbir brojeva* 2 *i* 6 *je paran* — *paran*$(2+6)$.

žena i paran označavaju svojstva, dok su majka i $+$ *označavaju funkcije.*

- U zapisu iskaza funkcije se označavaju funkcijskim simbolima.

Kvantifikacija

- U nekim slučajevima želimo da kažemo da svi objekti imaju neko svojstvo ili su u nekom odnosu.
- U nekim slučajevima želimo da kažemo da neki objekti imaju neko svojstvo ili su nekom odnosu.
- U slučaju da objekata o kojima govorimo ima konačno mnogo, moguće je napraviti konjunkciju ili disjunkciju iskaza koji govore o pojedinačnim objektima.
- Kako bi se izricanje ovakvih tvrdnji olakšalo uvode se promenljive i kvantifikatori.

Kvantifikacija

Primer

- *Svi ljudi su smrtni* — $\forall x.čovek(x) \Rightarrow smrtan(x)$.
- *Ne postoji savršen čovek* — $\neg\exists x.čovek(x) \wedge savršen(x)$.
- *Svi parni brojevi su deljivi sa 2* — $\forall n.paran(n) \Rightarrow 2 \mid n$.
- *Sledbenici nekih parnih brojeva su deljivi sa 3* — $\exists n.paran(n) \wedge 3 \mid (n+1)$.

Kvantifikatori se primenjuju na promenljive koje učestvuju u izrazima ravnopravno sa simbolima konstanti.

Kvantifikacija prvog reda

- U okviru logike prvog reda, kvantifikacija se vrši samo po primitivnim objektima — funkcijski i relacijski simboli se ne mogu kvantifikovati. Npr. rečenica da neki objekat a ima sva moguća svojstva ($\forall p.p(a)$) nije rečenica logike prvog reda.
- Logike višeg reda dopuštaju kvantifikaciju funkcijskih (odnosno relacijskih) simbola.

Šta se dešava ukoliko se kvantifikator izostavi?

- Svi do sada posmatrani iskazi predstavljali su rečenice.
- Ima li smisla u izrazima koristiti promenljive koje nisu pod dejstvom kvantifikatora?
- U tom slučaju, istinitosna vrednost iskaza nije jednoznačno određena već zavisi od vrednosti koje promenljive uzimaju.

Primer

- *Istinitosna vrednost tvrđenja* $\forall x.\ x > 0$ *ne zavisi od vrednosti promenljive* x*.*
- *Istinitosna vrednost tvrđenja* $x > 0$ *zavisi od vrednosti koju promenljiva* x *predstavlja.*

Pregled

Jezik (signatura)

- U zapisu formula logike prvog reda, pored logičkih simbola, učestvuju:
  - simboli konstanti
  - funkcijski simboli
  - relacijski (predikatski) simboli
- Svakom simbolu je pridružena arnost — broj argumenata na koji se funkcijski ili relacijski simbol primenjuje.
- Simboli konstanti se mogu shvatiti kao funkcijski simboli arnosti 0.
- Atomički iskazi (iskazna slova) se mogu shvatiti kao relacijski simboli arnosti 0.
- Neki binarni funkcijski ili relacijski simboli se obično pišu infiksno (npr. umesto $+(3, 2)$, pišemo $3 + 2$).

Jezik (signatura)

Definicija (Jezik (signatura))

Jezik $\mathcal{L}$ čini skup funkcijskih simbola Σ, skup relacijskih simbola Π i funkcija $ar : (\Sigma \cup \Pi) \to \mathbb{N}$ koja svakom simbolu dodelju arnost.

Jezik — primeri

- Jezik je obično određen matematičkom teorijom koja se formalizuje.

Primer

- *Ukoliko se priča o aritmetici, jezik je obično* $(0, 1, +, -, *, <, \leq, \ldots)$. *Npr.* $0 < 1 + 1$.
- *Ukoliko se priča o geometriji, jezik je obično čisto relacijski* (*incidentno*, *između*, *podudarno*, ...)

Sintaksa

- Sintaksu logike prvog reda definišemo u nekoliko faza:
  - Termovi
  - Atomičke formule
  - Formule
- Sa semantičkog stanovišta, termovi označavaju objekte nekog domena, dok formule imaju istinitosne vrednosti (tačno, netačno).

Termovi

- Termovi (jezika $\mathcal{L}$) su izrazi izgrađeni primenom funkcijskih simbola na konstante i promenljive.

Definicija

Skup termova (jezika $\mathcal{L}$) je najmanji skup koji zadovoljava:

- *Svaka promenljiva je term.*
- *Ako je c simbol konstante (jezika $\mathcal{L}$), onda je c term.*
- *Ako su $t_1, \ldots, t_k$ termovi, i f funkcijski simbol (jezika $\mathcal{L}$) arnosti k, onda je i $f(t_1, \ldots, t_k)$ takođe term.*

Termovi — primeri

Primer

Neka je $\mathcal{L}$ jezik sa konstantnim simbolima a i b, funkcijskim simbolom f arnosti 2 i g arnosti 1.
Neki od termova jezika $\mathcal{L}$ su:

- $a,\ b,\ x,\ y$
- $f(a,a),\ f(x,b),\ g(a)$
- $f(x,g(a)),\ f(f(a,b),g(x)),\ \ldots$

Implementacija u funkcionalnom jeziku

Obično se u apstraktnoj sintaksi ne vrši provera ispravnosti terma (arnosti funkcija) već se takva provera vrši prilikom parsiranja (obrade konkretne sintakse).

```
datatype term = Var string
              | Fn string  "term list"
```

Atomičke formule

Atomičke formule se grade primenom relacijskih simbola na termove.

Definicija

Skup atomičkih formula (jezika $\mathcal{L}$) je najmanji skup koji zadovoljava:

- *logičke konstante ($\top$ i $\bot$) su atomičke formule*
- *iskazno slovo (relacijski simbol arnosti 0) je atomička formula*
- *ako je ρ relacijski simbol jezika $\mathcal{L}$ arnosti k, i $t_1, \ldots, t_k$ termovi (jezika $\mathcal{L}$), onda je $\rho(t_1, \ldots, t_k)$ atomička formula.*

Atomičke formule — primeri

Primer

- $0 < 1,\ 2 + x \geq 5,\ x = 3$
- $\mathit{paran}(3),\ 4 \mid f(x)$
- $\mathit{između}(A, B, C), \mathit{podudarno}(A, B, A_1, B_1)$
- $\rho(x \circ 0, 0)$

Implementacija u funkcionalnom jeziku

Slično, provera arnosti relacija se ne vidi u apstraktnoj sintaksi.

```
datatype atom = R string "term list"
```

Formule

Formule se grade od atomičkih formula primenom logičkih veznika i kvantifikatora.

Definicija

Skup formula je najmanji skup koji zadovoljava:

- *Atomičke formule su formule*
- *Ako je A formula onda je $\neg(A)$ formula*
- *Ako su A i B formule, onda su i $(A) \wedge (B)$, $(A) \vee (B)$, $(A) \Rightarrow (B)$, $(A) \Leftrightarrow (B)$ formule.*
- *Ako je A formula, a x promenljiva onda su i $\forall x.(A)$ i $\exists x.(A)$ formule.*

Skraćeni zapis

- Prilikom zapisa formula, usvaja se prioritet veznika i u skladu sa tim se mogu izostaviti zagrade.

Primer

$$\forall x.\ f(x) < 3 \wedge x = 5 \Rightarrow \exists y.\ g(y) = x$$

je skraćeni zapis za

$$\forall x.\ (((f(x) < 3) \wedge (x = 5)) \Rightarrow (\exists y.\ (g(y) = 7)))$$

- Uzastopni kvantifikatori se kondenzuju.

Primer

$$\forall xyz.\ \rho(f(x, y), z)$$

je skraćeni zapis za

$$\forall x.\ \forall y.\ \forall z.\ \rho(f(x, y), z)$$

Implementacija u funkcionalnom jeziku

```
datatype formula =
      TRUE
    | FALSE
    | Atom atom
    | Not  formula
    | And  formula formula (infixl "And" 100)
    | Or   formula formula (infixl "Or"  100)
    | Imp  formula formula (infixl "Imp" 100)
    | Iff  formula formula (infixl "Iff" 100)
    | Forall string formula
    | Exists string formula
```

Slobodna i vezana pojavljivanja promenljivih

Pojavljivanja promenljive su vezana ako su pod dejstvom nekog kvantifikatora, a slobodna inače.

Definicija

- *Svako pojavljivanje promenljive u okviru atomičke formule je slobodno.*
- *Sva slobodna pojavljivanja promenljvih u formuli A su slobodna i u $\neg A$.*
- *Sva slobodna pojavljivanja promenljvih u formulama A i B su slobodna i u $A \wedge B$, $A \vee B$, $A \Rightarrow B$ i $A \Leftrightarrow B$.*
- *Sva slobodna pojavljivanja promenljive različite od x u formuli A su slobodna i u $\forall x.\ A$ i $\exists x.\ A$.*

Sva pojavljivanja promenljive u formuli koja nisu slobodna su vezana. Pritom, za slobodna pojavljivanja promenljive x u formuli A kažemo da su u formuli $\forall x.\ A$ vezana kvantifikatorom $\forall x$ (analogno za $\exists x.\ A$)

Slobodna i vezana pojavljivanja — primer

Primer

- *U formuli* $p(x, y) \Rightarrow (\forall x.\ q(x))$, *prvo pojavljivanje promenljive* x *i pojavljivanje promenljive* y *su slobodna, dok je drugo pojavljivanje promenljive* x *vezano.*
- *U formuli* $\exists x.\ p(x) \wedge (\forall x.\ q(x))$, *oba pojavljivanja promenljive* x *su vezana, pri čemu je pojavljivanje u okviru* $p(x)$ *pod dejstvom univerzalnog, a pojavljivanje u okviru* $q(x)$ *pod dejstvom egzistencijalnog kvantifikatora.*

Slobodne promenljive — implementacija

```
primrec fvt where
  "fvt (Var x) = {x}"
| "fvt (Fn args) = Union (map fvt args)"

fun fv where
  "fv False = {}"
| "fv True = {}"
| "fv (Atom (R r args)) = Union (map fvt args)"
| "fv (Not p) = fv p"
| "fv (p And q) = (fv p) Un (fv q)"
| "fv (p Or q) = (fv p) Un (fv q)"
| "fv (p Imp q) = (fv p) Un (fv q)"
| "fv (p Iff q) = (fv p) Un (fv q)"
| "fv (Forall x f) = (fv f) - {x}"
| "fv (Exists x f) = (fv f) - {x}"
```

Rečenice. Bazne formule

Definicija

- *Formula je rečenica akko nema slobodnih promenljivih.*
- *Formula je bazna akko nema promenljivih.*

Interpretacija

Obično u matematici, kada se napiše formula, implicitno se podrazumeva na koji skup objekata se formula odnosi (koje objekte konstante označavaju, šta su domeni promenljivih, koje funkcije i koje relacije su označene relacijskim i funkcijskim simbolima).

Primer

$$\forall x.\ 6|x \Rightarrow \neg paran(x+1)$$

Podrazumeva se da 6 *označava prirodan broj šest,* 1 *označava prirodan broj* 1*, da* $+$ *označava funkciju sabiranja dva prirodna broja, da* $|$ *označava relaciju deljivosti na skupu prirodnih brojeva dok paran označava svojstvo parnosti prirodnih brojeva.*

Interpretacija — domeni

Primer

- *Formula* $\forall x.\ \exists y.y + 1 = x$ *je tačna ako je domen skup celih brojeva, a netačna ako je domen skup prirodnih brojeva.*
- *Formula* $\forall x\ y.\ x < y \Rightarrow \exists z.\ (x < z \wedge z < y)$ *je tačna ako je domen skup realnih brojeva a netačna ako je domen skup celih brojeva.*

$\mathcal{L}$-strukture (modeli)

Definicija

Neka je dat jezik $\mathcal{L}$. $\mathcal{L}$-strukturu (model) $\mathfrak{D}$ čini:

- *Neprazan skup objekata (domen) D*
- *Za svaki simbol konstante c, njegova interpretacija $c_{\mathfrak{D}} \in D$.*
- *Za svaki funkcijski simbol f arnosti k, njegova interpretacija $f_{\mathfrak{D}} : D^k \to D$.*
- *Za svaki relacijski simbol ρ arnosti k, njegova interpretacija $\rho_{\mathfrak{D}} \subseteq D^k$.*

Na dalje ćemo obično pretpostavljati da je jezik $\mathcal{L}$ fiksiran pa ćemo umesto $\mathcal{L}$ struktura govoriti samo struktura (ili model).

Totalnost funkcija

U definiciji modela, podrazumeva se da su sve funkcije totalne i jednoznačne, tj. da su njihove interpretacije jednoznačno definisane za sve vrednosti argumenata iz domena D.

Tipovi (sorte)

- Primetimo da se svi objekti u okviru formule interpretiraju u okviru jedinstvenog domena, tj. da se ne razlikuju tipovi (sorte) objekata.
- Postoje varijacije logike (višesortne logike, logike višeg reda, . . .) koje dopuštaju uvođenje tipova (sorti) i tada se objekti svakog pojedinačnog tipa uzimaju iz posebnog domena.

Primer

- *Za svake dve tačke postoji prava koja ih sadrži.*
- $\forall A : t.\ \forall B : t.\ \exists l : p.\ A \in l \wedge B \in l$
- $\forall A.\ \forall B.\ t(A) \wedge t(B) \Rightarrow \exists l.\ p(l) \wedge A \in l \wedge B \in l$

Istinitosna vrednost rečenica

- Istinitosna vrednost rečenica zavisi samo od domena i interpretacije simbola (tj. određena je jednoznačno strukturom).
- Definicija ide rekurzivno po strukturi formule.
- Problem: Nije dovoljno posmatrati samo istinitosnu vrednost rečenica, jer se uklanjanjem kvantifikatora dobija formula sa slobodnim promenljivim čija vrednost nije određena jednoznačno dok se ne odredi vrednost promenljive.
  Npr. vrednost formule $\forall x.\ x > 0$ zavisi od vrednosti formule $x > 0$, pri čemu je potrebno dodatno naglasiti koje vrednosti x se posmatraju.
- Zbog toga se posmatra vrednost formule u datoj strukturi $\mathfrak{D}$ za datu valuaciju promenljivih v.

Vrednost termova

Vrednost terma u strukturi $\mathfrak{D}$ pri valuaciji v, definiše se rekurzivno po strukturi terma.

Definicija

Vrednost terma

- *Ako je term t promenljiva x, onda je njegova vrednost vrednost promenljive x u valuaciji v, tj. $\mathfrak{D}_v(t) = v(x)$.*
- *Ako je term t konstantni simbol c, onda je njegova vrednost interpretacija simbola c u strukturi $\mathfrak{D}$, tj. $\mathfrak{D}_v(t) = c_{\mathfrak{D}}$.*
- *Ako je term t oblika $f(t_1, \ldots, t_k)$, onda je njegova vrednost jednaka rezultatu primene funkcije koja je interpretacija simbola f u strukturi $\mathfrak{D}$ na vrednosti termova $t_1, \ldots t_k$, tj. $\mathfrak{D}_v(t) = f_{\mathfrak{D}}(\mathfrak{D}_v(t_1), \ldots, \mathfrak{D}_v(t_k))$.*

Implementacija u funkcionalnom jeziku

```
fun termval where
  "termval (domain, func, pred) v (Var x) = v x"
| "termval (domain, func, pred) v (Fn f args) =
        (func f) (map (termval (domain, func, pred) v) args)"
```

Zadovoljenje (tačnost)

- Činjenicu da je formula F tačna u strukturi $\mathfrak{D}$ pri valuaciji v označavaćemo sa $(\mathfrak{D}, v) \models F$. Kažemo još i da valuacija v zadovoljava formulu F u strukturi $\mathfrak{D}$, da su struktura $\mathfrak{D}$ i valuacija v model formule F itd.
- Uslovi pod kojima važi $(\mathfrak{D}, v) \models F$ definišu se rekurzivno po strukturi formule.
- Opet su moguće različite (međusobno ekvivalentne) definicije.

- Atomička formula $\rho(t_1, \ldots, t_k)$ je tačna u strukturi $\mathfrak{D}$ pri valuaciji v (tj. $(\mathfrak{D}, v) \models \rho(t_1, \ldots, t_k)$) akko važi $(\mathfrak{D}_v(t_1), \ldots, \mathfrak{D}_v(t_k)) \in \rho_{\mathfrak{D}}$, gde je $\mathfrak{D}_v(t)$ funkcija vrednosti terma u strukturi $\mathfrak{D}$ pri valuaciji v.
- Konstanta $\top$ je tačna u svakoj strukturi i valuaciji ($(\mathfrak{D}, v) \models \top$). Konstanta $\bot$ je netačna u svakoj strukturi i valuaciji ($(\mathfrak{D}, v) \not\models \bot$).
- Formula oblika $\neg F$ je tačna u strukturi $\mathfrak{D}$ pri valuaciji v akko je formula F netačna u strukturi $\mathfrak{D}$ pri valuaciji v (tj. $(\mathfrak{D}, v) \models \neg F$ akko $(\mathfrak{D}, v) \not\models F$).
- Formula oblika $F_1 \wedge F_2$ je tačna u strukturi $\mathfrak{D}$ pri valuaciji v akko su obe formule F_1 i F_2 tačne u strukturi $\mathfrak{D}$ pri valuaciji v (tj. $(\mathfrak{D}, v) \models F_1 \wedge F_2$ akko $(\mathfrak{D}, v) \models F_1$ i $(\mathfrak{D}, v) \models F_2$).

- Formula oblika $F_1 \vee F_2$ je tačna u strukturi $\mathfrak{D}$ pri valuaciji v akko je bar jedna od formula F_1 i F_2 tačna u strukturi $\mathfrak{D}$ pri valuaciji v (tj. $(\mathfrak{D}, v) \models F_1 \vee F_2$ akko $(\mathfrak{D}, v) \models F_1$ ili $(\mathfrak{D}, v) \models F_2$).
- Formula oblika $F_1 \Rightarrow F_2$ je tačna u strukturi $\mathfrak{D}$ pri valuaciji v akko su je formula F_1 netačna ili je formula F_2 tačna u strukturi $\mathfrak{D}$ pri valuaciji v (tj. $(\mathfrak{D}, v) \models F_1 \Rightarrow F_2$ akko $(\mathfrak{D}, v) \not\models F_1$ ili $(\mathfrak{D}, v) \models F_2$).
- Formula oblika $F_1 \Leftrightarrow F_2$ je tačna u strukturi $\mathfrak{D}$ pri valuaciji v akko su formule F_1 i F_2 istovremeno tačne ili istovremeno netačne u strukturi $\mathfrak{D}$ pri valuaciji v (tj. $(\mathfrak{D}, v) \models F_1 \Rightarrow F_2$ akko $(\mathfrak{D}, v) \models F_1$ i $(\mathfrak{D}, v) \models F_2$ ili $(\mathfrak{D}, v) \not\models F_1$ i $(\mathfrak{D}, v) \not\models F_2$).

- Formula oblika $\exists x.F$ je tačna u strukturi $\mathfrak{D}$ pri valuaciji v akko postoji valuacija v' dobijena od v samo izmenom vrednosti promenljive x takva da je F tačna u strukturi $\mathfrak{D}$ pri valuaciji v' (tj. $(\mathfrak{D}, v) \models \exists x.\ F$ ako postoji v' tako da $(\mathfrak{D}, v') \models F$). Drugim rečima, $(\mathfrak{D}, v) \models \exists x.\ F$ akko postoji element a iz domena D, tako da $(\mathfrak{D}, v(x \mapsto a)) \models F$.
- Formula oblika $\forall x.F$ je tačna u strukturi $\mathfrak{D}$ pri valuaciji v akko za svaku valuaciju v' dobijenu od v samo izmenom vrednosti promenljive x važi da je F tačna u strukturi $\mathfrak{D}$ pri valuaciji v' (tj. $(\mathfrak{D}, v) \models \forall x.\ F$ ako za svaku v' važi $(\mathfrak{D}, v') \models F$). Drugim rečima $(\mathfrak{D}, v) \models \forall x.\ F$ akko za svaki element a iz domena D, važi $(\mathfrak{D}, v(x \mapsto a)) \models F$.

Formalna definicija

```
  fun holds where
  "holds (domain, func, pred) v FALSE = False"
| "holds (domain, func, pred) v TRUE = True"
| "holds (domain, func, pred) v (Atom (R r args)) =
     (pred r) (map (termval (domain, func, pred) v) args)"
| "holds (domain, func, pred) v (Not F) =
     (~ holds (domain, func, pred) v F)"
| "holds (domain, func, pred) v (F1 And F2) =
     (holds (domain, func, pred) v F1 & holds (domain, func, pred) v F2)"
| "holds (domain, func, pred) v (F1 Or F2) =
     (holds (domain, func, pred) v F1 | holds (domain, func, pred) v F2)"
| "holds (domain, func, pred) v (F1 Imp F2) =
     (holds (domain, func, pred) v F1 -->  holds (domain, func, pred) v F2)"
| "holds (domain, func, pred) v (F1 Iff F2) =
     (holds (domain, func, pred) v F1 = holds (domain, func, pred) v F2)"
| "holds (domain, func, pred) v (Forall x p) =
     (ALL a : domain. holds (domain, func, pred) (v (x := a)) p)"
| "holds (domain, func, pred) v (Exists x p) =
     (EXISTS a : domain. holds (domain, func, pred) (v (x := a)) p)"
```

Problem određivanja tačnosti formule

- U iskaznoj logici, u trenutku kada je data valuacija, vrednost formule je (trivijalno) moguće odrediti.
- U logici prvog reda, iako je data valuacija i interpretacija, vrednost formule je komplikovano odrediti. Računarsku implementaciju na osnovu definicije je moguće odrediti samo u slučaju konačnih domena D (u slučaju beskonačnog domena, kod kvantifikacije bi bilo potrebno razmatrati vrednost potformule u besknačno mnogo različitih valuacija).

Tačnost rečenica ne zavisi od valuacije

Stav

Ako se valuacija v i v' poklapaju za sve slobodne promenljive formule F, onda $(\mathfrak{D}, v) \models F$ ako i samo ako $(\mathfrak{D}, v') \models F$.

Dokaz

Indukcijom po strukturi formule F.

- *Ako je F oblika $\top$ ili $\bot$, tvrđenje trivijalno važi.*
- *Ako je F oblika $\neg F'$, onda se skup slobodnih promenljivih za F i F' poklapa. $(\mathfrak{D}, v) \models F$ akko $(\mathfrak{D}, v) \not\models F'$. Po induktivnoj hipotezi, ovo važi akko $(\mathfrak{D}, v') \not\models F'$ akko $(\mathfrak{D}, v') \models F$.*

Tačnost rečenica ne zavisi od valuacije

Dokaz

- *Ako je F oblika $F' \wedge F''$, onda je skup slobodnih promenljivih formule F jednak uniji skupova slobodnih promenljivih formula F' i F'', pa se valuacije v i v' poklapaju i na skupovima slobodnih promenljivih formula F' i F'', pa se na ove formule sme primeniti induktivna hipoteza. $(\mathfrak{D}, v) \models F$ važi akko važi $(\mathfrak{D}, v) \models F'$ i $(\mathfrak{D}, v) \models F''$. Na osnovu i.h. ovo važi akko $(\mathfrak{D}, v') \models F'$ i $(\mathfrak{D}, v') \models F''$, a ovo važi akko $(\mathfrak{D}, v') \models F$.*
- *Ostali iskazni veznici se analogno razmatraju.*

Tačnost rečenica ne zavisi od valuacije

Dokaz

- *Ako je F oblika $\forall x.\ F'$, onda je skup slobodnih promenljivih formule F jednak skupu slobodnih promenljivih formule F', bez promenljive x. $(\mathfrak{D}, v) \models F$ važi akko za svako a iz D, $(\mathfrak{D}, v(x \mapsto a)) \models F'$. Valuacije $v(x \mapsto a)$ i $v'(x \mapsto a)$ se poklapaju na skupu slobodnih promenljivih formule F' te na osnovu i.h. $(\mathfrak{D}, v(x \mapsto a)) \models F'$ važi akko $(\mathfrak{D}, v'(x \mapsto a)) \models F'$, što važi akko $(\mathfrak{D}, v') \models F$.*
- *Slučaj kada je F oblika $\exists x.\ F'$ se razmatra analogno.*

Tačnost rečenica ne zavisi od valuacije

Posledica

Vrednost rečenica ne zavisi od valuacije, već zavisi samo od interpretacije nelogičkih simbola. Preciznije, za svake dve valuacije v i v' i rečenicu F, $(\mathfrak{D}, v) \models F$ važi ako i samo ako $(\mathfrak{D}, v') \models F$.

Semantika — primeri

Primer

Neka je dat jezik $\mathcal{L} = (0, 1, +, \cdot, =)$. *Posmatrajmo domen* $D = \mathbb{N}$. *Neka je struktura* $\mathfrak{D}_{\mathbb{N}}$ *određena na sledeći način:*

1. *Simbol* 0 *se interpretira prirodnim brojem* 0
2. *Simbol* 1 *se interpretira prirodnim brojem* 1
3. *Simbol* $+$ *se interpretira operacijom sabiranja prirodnih brojeva*
4. *Simbol* $\cdot$ *se interpretira operacijom množenja prirodnih brojeva*
5. *Simbol* $=$ *se interpretira jednakošću prirodnih brojeva*

Primer

Sledeće rečenice su tačne u $\mathfrak{D}_{\mathbb{N}}$*:*

- $\neg(0 = 1)$
- $\forall x.\ \neg(x + 1 = 0)$
- $\forall x\ y.\ x \cdot (y + 1) = x \cdot y + x$
- $\forall x.\ x = 0 \vee \neg(x = 0)$
- $\neg(\forall x.\ x = 0) \Leftrightarrow (\exists x.\ \neg(x = 0))$

Sledeće rečenice su netačne u $\mathfrak{D}_{\mathbb{N}}$*:*

- $\forall x.\ \exists y.\ y + 1 = x$
- $\forall x.\ \exists y.\ x + y = 0$

Vrednost formula koje nisu rečenice zavisi od valuacije:

- *Formula* $x + 1 = 1$ *je tačna u valuacijama u kojima je vrednost* x *prirodan broj nula, a netačna u valuacijama u kojima je vrednost* x *različita od prirodnog broja nula.*

Semantika — primeri

Primer

Neka je dat jezik $\mathcal{L} = (0, 1, +, \cdot, =)$. *Posmatrajmo domen* $D_{bool} = \{\top, \bot\}$. *Neka je struktura* $\mathfrak{D}_{bool}$ *određena na sledeći način:*

1. *simbol* 0 *se interpretira elementom* $\bot$,
2. *simbol* 1 *se interpretira elementom* $\top$,
3. *simbol* $+$ *se interpretira funkcijom koja elemente* x *i* y *iz* D_{bool} *slika u* $\top$ *akko je* $x \wedge y$ *tačna tj.*

$+$	$\top$	$\bot$
$\top$	$\top$	$\bot$
$\bot$	$\bot$	$\bot$

4. *simbol* $\cdot$ *se interpretira funkcijom koja elemente* x *i* y *iz* D_{bool} *slika u* $\top$ *akko je* $\neg(x \Leftrightarrow y)$ *tačna, tj.*

$\cdot$	$\top$	$\bot$
$\top$	$\bot$	$\top$
$\bot$	$\top$	$\bot$

5. *simbol* $=$ *se interpretira kao jednakost*

```
definition bool_interp where
"bool_interp == let
        domain = {True, False};
        funcs  = % f args. case (f, args) of
                    (''0'', []) => True |
                    (''1'', []) => False |
                    (''+'', [x, y]) => x & y |
                    (''*'', [x, y]) => ~(x = y);
        rels   =  % r args. case (r, args) of
                    (''='', [x::bool, y]) => x = y in
        (domain, funcs, rels)"
```

Semantika — primeri

Primer

Sledeće rečenice su tačne u $\mathfrak{D}_{bool}$.

- $\forall x.\ x = 0 \vee x = 1.$
- $\forall x.\ \exists y.\ x + y = 0.$
- $\forall x.\ x = 0 \vee \neg(x = 0)$
- $\neg(\forall x.\ x = 0) \Leftrightarrow (\exists x.\ \neg(x = 0))$

Sledeće rečenice su netačne u $\mathfrak{D}_{bool}$.

- $0 = 1$
- $\forall x.\ \neg(x + 1 = 0)$

Semantika — primeri

Primer

Da bi $\forall x.\ \exists y.\ x + y = 0$ *bila tačna u valuaciji* v*, potrebno je da:*

1. *formula* $\exists y.\ x + y = 0$ *bude tačna u valuaciji* $v(x \mapsto \top)$
2. *formula* $\exists y.\ x + y = 0$ *bude tačna u valuaciji* $v(x \mapsto \bot)$.

1. *Da bi formula* $\exists y.x + y = 0$ *bila tačna u valuaciji* $v(x \mapsto \top)$*, potrebno je da postoji element* d *iz* D *takav da je* $x + y = 0$ *tačna u valuaciji* $v(x \mapsto \top, y \mapsto d)$*. Zaista, formula* $x + y = 0$ *je tačna za* $d = \bot$*, tj. u valuaciji* $v' = v(x \mapsto \top, y \mapsto \bot)$*. Zaista, vrednost oba terma* $x + y$ *i* 0 *u* v' *je* $\bot$*, te su oni jednaki.*
2. *Da bi formula* $\exists y.x + y = 0$ *bila tačna u valuaciji* $v(x \mapsto \bot)$*, potrebno je da postoji element* d *iz* D *takav da je* $x + y = 0$ *tačna u valuaciji* $v(x \mapsto \bot, y \mapsto d)$*. Zaista, formula* $x + y = 0$ *je tačna za, npr.* $d = \bot$*, tj. u valuaciji* $v' = v(x \mapsto \top, y \mapsto \bot)$*. Zaista, vrednost oba terma* $x + y$ *i* 0 *u* v' *je* $\bot$*, te su oni jednaki.*

Semantika — primeri

Primer

Neka je dat jezik $\mathcal{L} = (0, 1, +, \cdot, =)$. *Za* $n \geq 2$, *posmatrajmo domen* $D_{mod_n} = \{0, 1, \ldots n-1\}$. *Neka je struktura* $\mathfrak{D}_{mod_n}$ *određena na sledeći način:*

1. *simbol* 0 *se interpretira brojem* 0,
2. *simbol* 1 *se interpretira brojem* 1,
3. *simbol* $+$ *se interpretira funkcijom koja sabira brojeve* x *i* y *iz* D *po modulu* n,
4. *simbol* $\cdot$ *se interpretira funkcijom koja množi elemente* x *i* y *po modulu* n,
5. *simbol* $=$ *se interpretira kao jednakost.*

Semantika — primeri

- Rečenica $\forall x.\ x = 0 \vee \neg(x = 0)$ je tačna u svim $\mathfrak{D}_{mod_n}$.
- Rečenica $\neg(\forall x.\ x = 0) \Leftrightarrow (\exists x.\ \neg(x = 0))$ je tačna u svim $\mathfrak{D}_{mod_n}$.
- Rečenica $\forall x.\ x = 0 \vee x = 1$ je tačna u $\mathfrak{D}_{mod_2}$, a netačna u $\mathfrak{D}_{mod_3}$.
- Rečenica $\forall x.\ \neg(x = 0) \Rightarrow \exists y.\ x \cdot y = 1$ je tačna u $\mathfrak{D}_{mod_n}$ akko je n prost broj.

Valjanost

- Pojam valjane formule donekle odgovara pojmu iskazne tautologije.

Definicija

Formula je valjana ako tačna u svakoj valuaciji, pri svakoj interpretaciji.

- Naravno, rečenica je valjana ako je tačna pri svakoj interpretaciji.

Stav

- *Rečenica je valjana akko je tačna pri svakoj interpretaciji.*
- *Formula F koja ima slobodne promenljive $x_1, \ldots x_n$ je valjana akko je valjano njeno univerzalno zatvorenje $\forall x_1, \ldots, x_n.\ F$.*

Valjane formule — primeri

Primer

- *Rečenica* $\forall x.\ x = 0 \vee \neg(x = 0)$ *je valjana.*
- *Rečenica* $\neg(\forall x.\ x = 0) \Leftrightarrow (\exists x.\ \neg(x = 0))$ *je valjana.*
- $(\forall x.P(x)) \Rightarrow P(a)$ *je valjana, dok formula* $P(x) \Rightarrow P(a)$ *ni formula* $\forall x.P(x) \Rightarrow P(a)$ *nije.*

Valjane formule — primeri

Primer (Paradoks pijanca)

Rečenica $\exists x.\ P(x) \Rightarrow \forall y.P(y)$ *je valjana.*
Time je tačna i sledeća interpretacija: ,,Postoji čovek, takav da ako on pije, onda svi piju".
Zaista, ako postoji čovek koji ne pije, on je taj traženi jer je u tom slučaju premisa netačna. Sa druge strane, ako svi piju, onda je konkluzija tačna pa bilo ko može biti traženi čovek (pretpostavka o nepraznosti domena je ključna kako bi se izbegao slučaj da nijedan čovek ne postoji).

Zadovoljivost

- Važno: definicije zadovoljivosti se značajno razlikuju po literaturi!
- U slučaju rečenica, stvar je jednostavna.

Definicija

Rečenica je zadovoljiva akko postoji interpretacija u kojoj je tačna.

- Problem nastaje prilikom definicije zadovoljivosti za formule sa slobodnim promenljima.
- Neke knjige izbegavaju definiciju zadovoljivosti za formule sa slobodnim promenljivim.
- Negde se (suštinski) daje naredna definicija:

Definicija

Formula je zadovoljiva akko postoji interpretacija u kojoj je tačna.

pri čemu je potrebno precizirati šta znači da je formula sa slobodnim promenljivim tačna (tj. zadovoljena) pri fiksiranoj interpretaciji.

Zadovoljenost (tačnost pri fiksiranoj interpretaciji).

Definicija

Formula F je zadovoljena interpretacijom $\mathfrak{D}$ akko postoji valuacija v tako da je formula tačna u valuaciji v pri interpretaciji $\mathfrak{D}$. Formula je zadovoljiva akko postoji interpretacija koja je zadovoljava.

Definicija

Formula F zadovoljena interpretacijom $\mathfrak{D}$ akko je za svaku valuaciju v formula tačna u valuaciji v pri interpretaciji $\mathfrak{D}$. Formula je zadovoljiva akko postoji interpretacija koja je zadovoljava.

Zadovoljivost (tačnost pri fiksiranoj interpretaciji)

Stav

U kontekstu prve definicije, formula F koja ima slobodne promenljive $x_1, \ldots x_n$ je zadovoljiva akko je zadovoljivo njeno egzistencijalno $\exists x_1, \ldots, x_n.\ F$.

Stav

U kontekstu druge definicije, formula F koja ima slobodne promenljive $x_1, \ldots x_n$ je zadovoljiva akko je zadovoljivo njeno univerzalno zatvorenje $\forall x_1, \ldots, x_n.\ F$.

Zadovoljenost (tačnost pri fiksiranoj interpretaciji)

Stav

Rečenica F je valjana akko je $\neg F$ nezadovoljiva.

U svetlu druge definicije, prethodni stav ne mora važi za formule koje nisu rečenice!

Primer

Formula $P(x) \vee \neg P(y)$ nije valjana (nije tačna u onoj valuaciji koja promenljivoj x dodeljuje objekat koji nema svojstvo P a promenljivoj y objekat koji ima svojstvo P) dok je njena negacija $\neg P(x) \wedge P(y)$ nezadovoljiva (nije tačna u valuacijama koje dodeljuju istu vrednost promenljivim x i y).

Logičke posledice

Definicija

Formula F je logička posledica skupa formula Γ (što označavamo sa $\Gamma \models F$) akko za svaku interpretaciju $\mathfrak{D}$ važi da ako $\mathfrak{D}$ zadovoljava svaku formulu iz Γ, onda zadovoljava i F (tj. ako je svaki model za skup Γ istovremeno i model za formulu F).

Primer

- *Rečenica $smrtan(Sokrat)$ je logička posledica rečenica $čovek(Sokrat)$ i $\forall x.\ čovek(x) \Rightarrow smrtan(x)$.*

Logičke posledice

Stav

Ako su $G_1, \ldots, G_n$ i F rečenice, onda $\{G_1, \ldots, G_n\} \models F$ akko je $G_1 \wedge \ldots \wedge G_n \Rightarrow F$ valjana.

U svetlu druge definicije, prethodni stav ne mora da važi ako nisu u pitanju rečenice!

Primer

Važi $P(x) \models P(a)$, ali formula $P(x) \Rightarrow P(a)$ nije valjana. Naime, ako $\mathfrak{D} \models P(x)$ onda za svaku valuaciju v, $P(x)$ važi u $\mathfrak{D}$. Samim tim, važi i za onu valuaciju koja x slika u onaj objekat koji je interpretacija konstante a. Sa druge strane, $P(x) \Rightarrow P(a)$ nije valjana jer je netačna u strukturi i valuaciji koja x-u dodeljuje različitu vrednost od objekta kojim je interpretirana konstanta a, pri čemu P ne važi za tu vrednost, dok važi za objekat kojim je interpretirana konstanta a.

Logička ekvivalentnost

Definicija

Formule F i G su logički ekvivalentne akko svaka interpretacija koja zadovoljava F zadovoljava i G i obratno.

Zamena

Definicija (Zamena promenljive u termu)

Term $t[x \to t']$ je zamena promenljive x u termu t termom t' ako je dobijen tako što se na mesto svakog pojavljivanja promenljive x u termu t postavi term t'.

Zamena promenljive u termu — implementacija

```
primrec tsubst x t' t where
  "tsubst x t' (Var x')  =
       (if x = x' then t' else t)"
| "tsubst x t' (Fn f args) =
       Fn f (map (tsubst x t') args)"
```

Zamena promenljive u termu — svojstva

Stav

Neka je v' valuacija dobijena od v tako što se promenljivoj x dodeljuje vrednost terma t' u valuaciji v, tj. $v' = v(x := \mathfrak{D}_v(t'))$. Onda je

$$\mathfrak{D}_v(t[x \to t']) = \mathfrak{D}_{v'}(t)$$

Zamena promenljive u formuli

- Postojanje slobodnih i vezanih promenljivih čini zamenu promenljive u okviru formule komplikovanijom operacijom.

Primer

- *Zamena promenljive x termom t ne bi trebalo da promeni formulu $\forall x.\ x = x$.*
- *Zamena promenljive x promenljivom y u formuli $\exists\ y.\ y + 1 = x$ je moguća tek nakon preimenovanja (tzv. α konverzije) vezane promenljive y, tj. vrednost zamene može biti formula $\exists y'.\ y' + 1 = y$. Direktna zamena dovodi do $\exists y.\ y + 1 = y$, što nije ono što se želi.*

Zamena promenljive u formuli — implementacija

```
primrec subst where
  subst x t False = False | subst x t True = True
| subst x t (Atom (R r args)) = Atom (R r (map (tsubst x t) args))
| subst x t (Not F) = Not (subst x t F)
| subst x t (F1 And F2) = (subst x t F1) And (subst x t F2)
| subst x t (F1 Or F2)  = (subst x t F1) Or (subst x t F2)
| subst x t (F1 Imp F2) = (subst x t F1) Imp (subst x t F2)
| subst x t (F1 Iff F2) = (subst x t F1) Iff (subst x t F2)
| subst x t (ALL a F) = substquant x t ALL a F
| subst x t (EX a F) = substquant x t EX a F

definition substquant where
"substquant x t Q a F ==
   if x = a then Q a F
   else let a' = fresh_var x t a F in
      Q a' (subst x t (subst a a' F))"
```

Zamena promenljive u formuli — primeri

Primer

- $(\forall x.\ P(x))[x \to y] = \forall x.P(x)$
- $(\forall z.\ P(x,z))[x \to y] = \forall z'.P(y,z')$ – *preimenovanje nije neophodno u ovom primeru*
- $(\forall y.\ P(x,y))[x \to y] = \forall y'.P(y,y')$ – *preimenovanje jeste neophodno u ovom primeru*

Broj preimenovanja se može smanjiti. Jedan od kriterijuma za ispitivanje da li je neophodno vršiti preimenovanje je da li term t sadrži kvantifikovanu promenljivu

Zamena promenljive u formuli — svojstva

Stav

Neka je v' valuacija dobijena od v tako što se promenljivoj x dodeljuje vrednost terma t u valuaciji v, tj. $v' = v(x := \mathfrak{D}_v(t))$. Onda je

$$(D, v) \vDash F[x \to t] \ akko \ (D, v') \vDash F$$

Zamene formule u formuli

Pored zamene promenljive u formuli, moguće je definisati i zamenu potformule u okviru formule.

Primer

Rezultat zamene formule $\neg(P(x) \vee Q(x))$ *formulom* $\neg P(x) \wedge \neg Q(x)$ *u formuli* $\forall x.\ \neg(P(x) \vee Q(x)) \wedge R(x)$ *je formula* $\forall x.\ (\neg P(x) \wedge \neg Q(x)) \wedge R(x)$.

Zamene iskaznih slova formulama logike prvog reda

Takođe, moguće je (čisto sintaksno) definisati i zamene iskaznih slova u iskaznim formulama

Primer

Rezultat zamene P *sa* $p(x)$ *i* Q *sa* $\forall y.\ y > 0$ *u formuli* $P \vee Q$ *je* $p(x) \vee (\forall y.\ y > 0)$.

Pregled

Negaciona normalna forma

Negaciona normalna forma se definiše po uzoru na iskaznu logiku.

Definicija

Formula je u negacionoj normalnoj formi (NNF) akko je izgrađena od literala (prvog reda) korišćenjem isključivo veznika $\wedge$ i $\vee$ i kvantifikatora ili je logička konstanta ($\top$ ili $\bot$).

Pri svođenju date formule na NNF, pored logičkih zakona nasleđenih iz iskazne logike koriste se i zakoni:

$$\neg\forall x.\ A \equiv \exists x.\ \neg A$$

$$\neg\exists x.\ A \equiv \forall x.\ \neg A$$

Prenex normalna forma

Definicija

Formula je prenex normalnoj formi ako je oblika

$$Q_1\ x_1.\ \ldots\ Q_n\ x_n.\ F,$$

pri čemu su Q_i kvantifikatori $\forall$ ili $\exists$, a formula F ne sadrži kvantifikatore.

Formula koja je u NNF se može prevesti u prenex normalnu formu primenama sledećih ekvivalencija:

$$
\begin{array}{llll}
(\forall x.A) \wedge B & \equiv & (\forall x.\ A \wedge B) & (\exists x.A) \wedge B \equiv (\exists x.\ A \wedge B) \\
(\forall x.A) \vee B & \equiv & (\forall x.\ A \vee B) & (\exists x.A) \vee B \equiv (\exists x.\ A \vee B) \\
B \wedge (\forall x.A) & \equiv & (\forall x.\ B \wedge A) & B \wedge (\exists x.A) \equiv (\exists x.\ B \wedge A) \\
B \vee (\forall x.A) & \equiv & (\forall x.\ B \vee A) & B \vee (\exists x.A) \equiv (\exists x.\ B \vee A)
\end{array}
$$

pri čemu, ako se promenljiva x javlja slobodna u B, potrebno je izvršiti njeno preimenovanje (α-konverziju) u formuli $\forall x.\ A$.

Skraćenice

U nekim slučajevima, moguće je koristiti i naredna pravila koja smanjuju broj kvantifikatora.

$$(\exists x.\ A) \vee (\exists x.B) \equiv (\exists x.A \vee B)$$
$$(\forall x.\ A) \wedge (\forall x.B) \equiv (\forall x.A \wedge B)$$

Obratiti pažnju da naredna pravila ne predstavljaju ekvivalencije i ne mogu se koristiti za prevođenje u prenex normalnu formu.

$$(\exists x.\ A) \wedge (\exists x.B) \equiv (\exists x.A \wedge B)$$
$$(\forall x.\ A) \vee (\forall x.B) \equiv (\forall x.A \vee B)$$

Teorema Prenex

Teorema

Za svaku formulu postoji formula koja je u prenex normalnoj formi i koja joj je ekvivalentna.

Skolemizacija

Definicija

Formula je u Skolemovoj normalnoj formi ukoliko je oblika

$$\forall x_1. \ \ldots \ \forall x_n. \ F,$$

Postupak svođenja u Skolemovu normalnu formu naziva se skolemizacija.
Obično (mada ne i obavezno) se skolemizacija primenjuje na formule koje su u prenex normalnoj formi.

Ekvizadovoljivost pri skolemizaciji

- Osnovni korak skolemizacije je uklanjanje egzistencijalnih kvantifikatora.
- Egzistencijalni kvantifikatori se ne mogu ukloniti, a da se zadrži ekvivalentnost.
- Ipak, moguće je ukloniti egzistencijalne kvantifikatore, i zadržati zadovoljivost, što je za većinu primena dovoljno.

Skolemizacija — primeri

Primer

Posmatrajmo formulu $\exists x.\ p(x)$ jezika $\mathcal{L} = \{p\}$ i formulu $p(c)$, gde je c novi simbol konstante (ne pripada jeziku $\mathcal{L}$). Neka je $\mathcal{L}' = \mathcal{L} \cup \{c\}$.

- *Ukoliko je rečenica $\exists x.\ p(x)$ zadovoljiva, postoji $\mathcal{L}$-struktura $\mathfrak{D}$ takva da $\mathfrak{D} \models \exists x.\ p(x)$. To znači da u domenu D postoji element $\hat{x}$ takav važi $p_{\mathfrak{D}}(\hat{x})$. Ukoliko posmatramo $\mathcal{L}'$-strukturu $\mathfrak{D}'$ dobijenu od $\mathfrak{D}$ dodatnim interpretiranjem simbola c elementom $\hat{x}$ (tj. $c_{\mathfrak{D}'} = \hat{x}$), važi $\mathfrak{D}' \models p(c)$, te je i rečenica $p(c)$ zadovoljiva.*
- *Ukoliko je $\mathfrak{D}'$ $\mathcal{L}'$-struktura takva da važi $\mathfrak{D}' \models p(c)$, tada važi $p(c_{\mathfrak{D}'})$, te važi da $\mathfrak{D}' \models \exists x.\ p(x)$.*

Skolemizacija — primeri

Primer

Posmatrajmo formulu $\forall x.\ \exists y.\ p(x, y)$ *jezika* $\mathcal{L} = \{p\}$ *i formulu* $\forall x.\ p(x, f(x))$, *gde je* f *novi funkcijski simbol (ne pripada jeziku* $\mathcal{L}$*). Neka je* $\mathcal{L}' = \mathcal{L} \cup \{f\}$.

- *Ukoliko je rečenica* $\forall x.\ \exists y.\ p(x, y)$ *zadovoljiva, postoji* $\mathcal{L}$*-struktura* $\mathfrak{D}$ *takva da za svaki element* $\hat{x} \in D$ *postoji element* $\hat{y} \in D$ *takav važi* $p_{\mathfrak{D}}(\hat{x}, \hat{y})$. *Dakle (pod pretpostavkom aksiome izbora), postoji funkcija* $\hat{f} : D \to D$ *koja elementima* $\hat{x}$ *dodeljuje odgovarajuće* $\hat{y}$. *Ukoliko posmatramo* $\mathcal{L}'$*-strukturu* $\mathfrak{D}'$ *dobijenu od* $\mathfrak{D}$ *dodatnim interpretiranjem simbola* f *funkcijom* $\hat{f}$ *(tj.* $f_{\mathfrak{D}'} = \hat{f}$*), važi* $\mathfrak{D}' \vDash \forall x.p(x, f(x))$, *te je ova rečenica zadovoljiva.*
- *Ukoliko je* $\mathfrak{D}'$ $\mathcal{L}'$*-struktura takva da važi* $\mathfrak{D}' \vDash \forall x.\ p(x, f(x))$, *tada za svaki element* $\hat{x} \in D$ *važi* $p(\hat{x}, f_{\mathfrak{D}'}(\hat{x}))$, *te važi da* $\mathfrak{D}' \vDash \forall x.\ \exists y.\ p(x, y)$.

Postupak skolemizacije

Procedura skolemizacije se zasniva na uzastopnoj primeni sledećih transformacija:

- Ako je formula oblika $\exists x.A$, bira se novi simbol konstante c (koji ne pripada jeziku $\mathcal{L}$) i formula se zamenjuje formulom $A[x \to c]$.
- Ako je formula oblika $\forall x_1 \ldots \forall x_n.\exists x.\ A$, bira se novi funkcijski simbol f (koji ne pripada jeziku $\mathcal{L}$ i formula se zamenjuje formulom $A[x \to f(x_1, \ldots, x_n)]$.

Skolemizacija i logika višeg reda

Naredna ekvivalencija važi u logici drugog reda (u logici prvog reda nije dopuštena kvantifikacija nad funkcijama):

$$\forall x.\ \exists y.\ p(x, y) \equiv \exists f.\ \forall x.\ p(x, f(x))$$

Opštije

$$\forall x_1 \ldots \forall x_n.\ \exists y.\ p(x_1, \ldots, x_n, y) \equiv \exists f.\ \forall x_1 \ldots \forall\, x_n.\ p(x_1, \ldots, x_n, f(x_1, \ldots, x_n))$$

Teorema o skolemizaciji

Teorema

Formula dobijena skolemizacijom formule F je ekvizadovoljiva formuli F.

Dokaz

Teorema se dokazuje na način sličan opisanom u prethodna dva primera.

Oslobađanje univerzalnih kvantifikatora?

- Da li je moguće ukloniti univerzalne kvantifikatore tako da formula ostane ekvizadovoljiva?
- Zavisi od definicije zadovoljive formule.
- Ukoliko se koristi prva definicija (postoji interpretacija i postoji valuacija) nije moguće osloboditi se univerzalnih kvantifikatora.
- Ukoliko se koristi druga definicija (postoji interpretacija tako da je za svaku valuaciju) univerzalni kvantifikatori se mogu samo izostaviti i dobija se ekvizadovoljiva formula.

Formule bez kvantifikatora

Definicija

Formula je formula bez kvantifikatora (eng. quantifier free) ukoliko je oblika

$$\forall x_1. \ \ldots \ \forall x_n. \ F.$$

Često se, po konvenciji, za ove formule navodi samo matrica F, dok se blok kvantifikatora izostavlja — ovo je naročito opravdano u svetlu druge definicije zadovoljivosti.

Stav

Za svaku formulu postoji formula bez kvantifikatora koja joj je ekvizadovoljiva.

Prethodni stav ne važi za valjanost.

Definicija

- *KNF i DNF formula bez kvantifikatora logike prvog reda se definišu analogno iskaznoj logici.*
- *Formula je u klauzalnoj formi ukoliko je oblika*

$$\forall x_1. \ \ldots \ \forall x_n. \ F,$$

pri čemu je F u KNF.

Teorema o klauzalnoj formi

Teorema

Za svaku formulu postoji formula u klauzalnoj formi koja joj je ekvizadovoljiva.

Dokaz

Na formulu se primeni NNF, Prenex, Skolemizacija i konverzija matrice u KNF (npr. Cajtinovom transformacijom).

Dokazivanje pobijanjem

- Jedan od pristupa dokazivanja valjanosti neke rečenice je dokazivanje pobijanjem koje podrazumeva dokazivanje nezadovoljivosti negacije polazne rečenice.
- Klauzalna forma se najčešće koristi u svetlu dokazivanja pobijanjem.

Pregled

Erbranova teorema

- Erbran (fr. Herbrand) 1930. — tragovi u ranijim radovima Skolema i Gedela
- U logici prvog reda nije odlučivo (ne postoji opšti postupak) da li je neka formula valjanja (Entscheidungsproblem, postavio Hilbert, negativno razrešili Čerč i Tjuring).
- Međutim, Erbranova teorema daje jedan od načina da se pokaže poluodlučivost: postoji algoritam koji za svaku valjanu formulu može da pokaže da je valjana.
- Erbranova teorema uspostavlja veze između logike prvog reda i iskazne logike.

Iskazni pogled na formule bez kvantifikatora

Formule bez kvantifikatora se mogu tretirati kao iskazne formule (gde se umesto iskaznih slova koriste atomi logike prvog reda).

Primer

Posmatrajmo formulu $x > 0 \vee \neg(x > 0)$ *kao iskaznu formulu nad skupom atoma* $\{x > 0\}$. *Iskazne valuacije proglašavaju ovaj atom ili za tačan ili za netačan. U oba slučaja polazna formula je tačna pa je iskazna tautologija. Sa druge strane, formula* $x > 0 \vee \neg(x > 0)$ *je i valjana (u logici prvog reda). Zaista, za svaku valuaciju* v *i svaku interpretaciju* $\mathfrak{D}$ *važi da* $(\mathfrak{D}, v) \vDash x > 0 \vee \neg(x > 0)$. *Valuacija* v *definiše vrednost atomičke formule* $x > 0$, *međutim, kakva god ona bila, na osnovu semantike logike prvog reda, sledi da je formula tačna.*
Formula $p \vee \neg p$ *je iskazna tautologija, a prethodna formula se može dobiti zamenom promenljive u tautologiji. Ovakve formule se nazivaju i izvodima tautologija (ili tautologijama prvog reda).*

Veza između valjanosti i (iskazne) tautologičnosti za formule bez kvantifikatora

Stav

Formula bez kvantifikatora je valjana akko je iskazna tautologija.

Dokaz

Jednostavno se dokazuje da, ako je formula bez kvantifikatora iskazna tautologija, onda je ona valjana (indukcijom po strukturi formule prateći definicije semantike u iskaznoj logici i logici prvog reda).

Dokaz

Drugi smer je komplikovaniji. Pretpostavimo da je formula bez kvantifikatora F valjana i pokažimo da je iskazna tautologija. Fiksirajmo iskaznu valuaciju $\bar{v}$. Pokazaćemo da je moguće izgraditi $\mathcal{L}$-strukturu $\mathcal{D}$ i valuaciju prvog reda v tako da je $\bar{v} \models F$ akko važi $(\mathfrak{D}, v) \models F$. Dovoljno je pokazati da tvrđenje važi za svaki atom $R(t_1, \ldots, t_n)$ formule F, tj. da važi $\bar{v} \models R(t_1, \ldots, t_n)$ akko $(\mathfrak{D}, v) \models R(t_1, \ldots, t_n)$ i nastavak dokaza sledi indukcijom.

Dokaz

Model $\mathfrak{D}$ gradimo od sintaksnog materijala, tj. domen D je skup termova jezika $\mathcal{L}$. Interpretacije simbola konstanti su oni sami (tj. $c_{\mathfrak{D}} = c$), a interpretacije funkcijskih simbola se grade tako da je $f_{\mathfrak{D}}(t_1, \ldots t_n) = f(t_1, \ldots, t_n)$. Vrednost svakog terma u ovako definisanoj valuaciji je on sâm (tj. $\mathfrak{D}_v(t) = t$). Dakle, različiti termovi imaju obavezno različite vrednosti. Sada je moguće definisati interpretaciju svakog relacijskog simbola R tako što je $R_{\mathfrak{D}}(t_1, \ldots, t_n)$ tačno akko $\bar{v} \vDash R(t_1, \ldots, t_n)$.

Neka valuacija v dodeljuje svakoj promenljivoj nju samu (tj. $v(x) = x$).

Ovako definisani $\mathfrak{D}$ i v zadovoljavaju traženo svojstvo. Zaista $(\mathfrak{D}, v) \vDash R(t_1, \ldots, t_n)$ akko $R_{\mathfrak{D}}(\mathfrak{D}_v(t_1), \ldots, \mathfrak{D}_v(t_n))$ akko $R_{\mathfrak{D}}(t_1, \ldots, t_n)$ akko $\bar{v} \vDash R(t_1, \ldots, t_n)$.

Tvrđenje dalje jednostavno sledi indukcijom po strukturi formule F (prateći definicije semantike u iskaznoj logici i logici prvog reda).

Kanonski modeli

- Termovski modeli izgrađeni na način kao u prethodnom dokazu se veoma često koristi.
- Ovakvi modeli nazivaju se *kanonski modeli (kanonske interpretacije)*.

Veza između zadovoljivosti i iskazne zadovoljivosti formula bez kvantifikatora

- Kod formula bez kvantifikatora, valjanosti u logici prvog odgovara iskazna tautologičnost, međutim, važnije pitanje je da li postoji slična veza za zadovoljivost (svakoj formuli možemo pridružiti ekvizadovoljivu formulu bez kvantifikatora, a ne ekvivaljanu).
- Za rečenice, tj. bazne formule (formule bez promenljivih), odnos je jednostavan.

Stav

Bazna formula bez kvantifikatora je zadovoljiva akko je iskazno zadovoljiva.

- Za formule sa slobodnim (univerzalno kvantifikovanim) promenljivim, odnos nije tako jednostavan.

Veza između zadovoljivosti i iskazne zadovoljivosti formula bez kvantifikatora

- Jedan smer je i dalje trivijalan.

Stav

Ako je formula bez kvantifikatora zadovoljiva, ona je i iskazno zadovoljiva.

Dokaz

Ako F ne bi bila iskazno zadovoljiva, onda bi $\neg F$ bila tautologija i zato bi bila valjana formula pa F ne bi mogla biti zadovoljiva.

Veza između zadovoljivosti i iskazne zadovoljivosti formula bez kvantifikatora

- Drugi smer ne važi.

Primer

Formula $P(x) \wedge \neg P(y)$ *je iskazno zadovoljiva u valuaciji u kojoj je* $P(x)$ *tačno, a* $P(y)$ *netačno. Međutim formula* $\forall x.\ \forall y.P(x) \wedge \neg P(y)$ *nije zadovoljiva formula logike prvog reda.*

Veza između zadovoljivosti i iskazne zadovoljivosti formula bez kvantifikatora

- Formule bez kvantifikatora su u suštini univerzalno kvantifikovane.
- Univerzalno kvantifikovane formule predstavljaju suštinski „beskonačne konjunkcije" baznih formula.
- Npr. $\forall x.\ p_1(x, c_1) \vee p_2(f(x), c_2)$ predstavlja konjunkciju

  $(p_1(a_1, c_1) \vee p_2(f(a_1), c_2)) \wedge (p_1(a_2, c_1) \vee p_2(f(a_2), c_2)) \wedge \ldots,$

  gde a_1, a_2, ... predstavljaju različite elemente domena.

Veza između zadovoljivosti i iskazne zadovoljivosti formula bez kvantifikatora

- Cilj je ispitati (ne)zadovoljivost formule.
- Ispostaviće se da je u tu svrhu, umesto proizvoljnih domena, dovoljno posmatrati (u neku ruku najopštije) kanonske modele.
- U prethodnom primeru, elementi kanonskog modela su identifikovani termovima c_1, c_2, $f(c_1)$, $f(c_2)$, $f(f(c_1))$, $f(f(c_2))$, ...
- Formula $\forall x.\ p_1(x, c_1) \vee p_2(f(x), c_2)$ predstavlja konjunkciju

$$\begin{aligned} (p_1(c_1, c_1) \vee p_2(f(c_1), c_2)) \quad & \wedge \\ (p_1(c_2, c_1) \vee p_2(f(c_2), c_2)) \quad & \wedge \\ (p_1(f(c_1), c_1) \vee p_2(f(f(c_1)), c_2)) \quad & \wedge \quad \ldots \end{aligned}$$

Erbranov univerzum

- Domen kanonskih interpretacija je bilo koji skup termova koji sadrži sve konstante jezika $\mathcal{L}$ i zatvoren je za sve funkcijske simbole jezika $\mathcal{L}$.
- Iz praktičnih razloga, poželjno je koristiti što manje domene.
- Pokazaće se da je za ispitivanje zadovoljivosti dovoljno posmatrati skup svih baznih termova jezika $\mathcal{L}$.

Erbranov univerzum

- Skup svih baznih termova jezika $\mathcal{L}$ nazivamo Erbranov univerzum i označavamo sa $H(\mathcal{L})$.
- Ukoliko jezik ne sadrži ni jednu konstantu, u njega se veštački umeće novi simbol konstante (npr. c) kako Erbranov unverzum ne bi bio prazan.

Primer

Ako je jezik $\mathcal{L} = \{c, f\}$, *onda je Erbranov univerzum* $H = \{c, f(c), f(f(c)), \ldots\}$. *Ako je jezik* $\mathcal{L} = \{+, \cdot\}$, *onda je Erbranov univerzum*
$\{c, c + c, c \cdot c, c + c + c, c + c \cdot c, c \cdot c + c, c \cdot c \cdot c, \ldots\}$.

Erbranov univerzum formule i njene bazne instance

- Erbranov univerzum formule $H(F)$ je Erbranov univerzum jezika sačinjenog od simbola koji se javljaju u toj formuli.
- Bazne instance formule, dobijaju se kao rezultat zamene promenljivih elementima njenog Erbranovog univerzuma.

Veza između zadovoljivosti i iskazne zadovoljivosti formula bez kvantifikatora

Ključni kriterijum kojim se uspostavlja veza između zadovoljivosti u logici prvog reda i iskazne zadovoljivosti za formule bez kvantifikatora dat je narednom teoremom:

Teorema (Erbran)

Formula bez kvantifikatora F je zadovoljiva akko je skup svih njenih baznih instanci iskazno zadovoljiv. Preciznije, formula F oblika $\forall x_1 \ \ldots \ x_n.\phi(x_1, \ldots, x_n)$ je zadovoljiva akko je zadovoljiv skup $\{\phi[x_1 \to t_1] \ldots [x_n \to t_n] \mid t_1, \ldots, t_n \in H(F)\}$.

Dokaz

Radi pojednostavljivanja zapisa, bazne instance $\phi[x_1 \to t_1] \ldots [x_n \to t_n]$ *ćemo kraće označavati sa* $F[x_i \to t_i]$. *Ako je* F *zadovoljiva ona važi u nekom modelu* $\mathfrak{D}$ *i u svakoj valuaciji* v. *Neka je iskazna valuacija* $\bar{v}$ *takva da za svaki atom* a *važi* $\bar{v} \vDash a$ *akko* $(\mathfrak{D}, v) \vDash a$. *Pokažimo da* $\bar{v}$ *zadovoljava sve bazne instance formule* F.
Neka je i proizvoljna instancijacija, tj. preslikavanje promenljivih x_i *u bazne termove* t_i *iz* H. *Posmatrajmo baznu instancu* $F[x_i \to t_i]$. *Na osnovu definicije* $\bar{v}$, *važi da* $\bar{v} \vDash F[x_i \to t_i]$ *akko* $(\mathfrak{D}, v) \vDash F[x_i \to t_i]$. *Neka je* v' *valuacija dobijena od* v *tako što se svakoj promenljivoj* x_i *dodeljuje vrednost terma* t_i *u valuaciji* v. *Tada* $(\mathfrak{D}, v) \vDash F[x_i \to t_i]$ *akko* $(\mathfrak{D}, v') \vDash F$. *Međutim, ovo je tačno jer je* $\mathfrak{D}$ *model za* F, *a vrednost* F *ne zavisi od valuacije.*

Dokaz

Obratno, neka valuacija $\bar{v}$ zadovoljava sve bazne instance formule F. Neka je $\mathfrak{D}$ kanonska interpretacija nad Erbranovim univerzumom određena valuacijom $\bar{v}$ i neka je v proizvoljna valuacija prvog reda koja preslikava promenljive u elemente Erbranovog univerzuma. Potrebno je pokazati da $(\mathfrak{D}, v) \models F$. Pokažimo da važi $(\mathfrak{D}, v) \models F$ akko $\bar{v} \models F[x_i \to v(x_i)]$ (tvrđenje odatle sledi jer je $F[x_i \to v(x_i)]$ bazna instanca koja je tačna u $\bar{v}$). Dovoljno je dokazati tvrđenje za atomičke formule $R(t_1, \ldots, t_n)$ (dokaz dalje sledi indukcijom po strukturi formule F). Važi:
$(\mathfrak{D}, v) \models R(t_1, \ldots, t_n)$ *akko* $R_{\mathfrak{D}}(\mathfrak{D}_v(t_1), \ldots, \mathfrak{D}_v(t_n))$ *akko*
$R_{\mathfrak{D}}(t_1[x_i \to v(x_i)], \ldots, t_n[x_i \to v(x_i)])$ *akko*
$\bar{v} \models R(t_1[x_i \to v(x_i)], \ldots, t_n[x_i \to v(x_i)])$ *akko*
$\bar{v} \models R(t_1, \ldots, t_n)[x_i \to v(x_i)]$.

Teorema (Erbran)

Erbranova interpetacija $\mathfrak{D}$ zadovoljava formulu bez kvantifikatora F akko zadovoljava sve njene bazne instance.

Dokaz

Pretpostavimo da $\mathfrak{D}$ zadovoljava F. Neka je v proizvoljna valuacija prvog reda, tj. preslikavanje promenljivih x_i u bazne termove t_i iz H. Posmatrajmo baznu instancu $F[x_i \to t_i]$. Neka je v' valuacija dobijena od v tako što se svakoj promenljivoj x_i dodeljuje term t_i. Tada $(\mathfrak{D}, v) \models F[x_i \to t_i]$ akko $(\mathfrak{D}, v') \models F$. Međutim, ovo je tačno jer je $\mathfrak{D}$ model za F, a vrednost F ne zavisi od valuacije.

Dokaz

Obratno, neka $\mathfrak{D}$ zadovoljava sve bazne instance. Neka je v proizvoljna valuacija koja preslikava promenljive x_v u termove t_v Erbranovog univerzuma. Neka je v' valuacija koja svakoj promenljivoj x_v dodeljuje vrednost terma t_v u $\mathfrak{D}$, tj. $\mathfrak{D}_v(t_v)$. Međutim, pošto je $\mathfrak{D}$ kanonska, onda je $\mathfrak{D}_v(t_v) = t_v$ pa je $v' = v$. Dakle, važi $(\mathfrak{D}, v) \models F$ akko $(\mathfrak{D}, v') \models F$ akko $(\mathfrak{D}, v) \models F[x_v \to t_v]$. Tvrđenje dalje sledi jer je $F[x_v \to t_v]$ bazna instanca i tačna je u $\mathfrak{D}$.

Mehanizacija Erbranove teoreme

- Ispitivanje zadovoljivosti se svodi na ispitivanje zadovoljivosti skupa svih baznih instanci.
- Ovaj skup može biti beskonačan.
- Ipak, kompaktnost obezbeđuje potrebnu konačnost

Teorema

Formula bez kvantifikatora je nezadovoljiva akko postoji konačan nezadovoljiv podskup njenih baznih instanci.

- Procedure Erbranovog tipa nabrajaju skupove baznih instanci dodajući nove instance.
  - Ukoliko se utvrdi iskazna nezadovoljivost tekućeg skupa instanci, polazna formula je nezadovoljiva.
  - Ukoliko se iscrpe sve bazne instance, a nezadovoljivost se ne utvrdi, polazna formula je zadovoljiva.
  - Moguće je da procedura beskonačno dodaje nove bazne instance (u slučaju da je polazna formula zadovoljiva) i da se ne zaustavlja.
- Najčuvenija je Gilmorova procedura (prva implementirana) koja instance prevodi u DNF i tako ispituje njihovu zadovoljivost.

Erbranova teorema — primeri

Primer

Pokažimo da je rečenica $\exists x.\ \forall y. P(x,y) \Rightarrow \forall y.\ \exists x.\ P(x,y)$ *valjana. Pobijanjem, potrebno je pokazati da je njena negacija nezadovoljiva. Nakon NNF transformacije, dobija se formula* $\exists x.\ \forall y.\ P(x,y) \wedge \exists y.\ \forall x. \neg P(x,y)$. *Prenex trasformacija daje njoj ekvivalentnu formulu* $\exists z. \exists u. \forall w.\ P(z,w) \wedge \neg P(w,u)$. *Skolemizacijom se dobija formula bez kvantifikatora* $\forall w.\ P(c_1,w) \wedge \neg P(w,c_2)$ *koja je zadovoljiva akko je polazna formula valjana.*
Na osnovu Erbranove teoreme, prethodna formula je zadovoljiva akko je iskazno zadovoljiva formula

$$P(c_1,c_1) \wedge \neg P(c_1,c_2) \wedge P(c_1,c_2) \wedge \neg P(c_2,c_2),$$

što nije slučaj.

Primer

Paradoks berberina: Nije moguće da postoji berberin koji brije sve one koji sami sebe ne briju:

$$\neg(\exists b.\ \forall x.\ brije(b, x) \Leftrightarrow \neg brije(x, x))$$

Da bi se pokazalo da je prethodna formula valjana, pokažimo da je njena negacija nezadovoljiva. Nakon svođenja u normalnu formu, dobija se:

$$\forall x.\ (\neg brije(c, x) \vee \neg brije(x, x)) \wedge (brije(c, x) \vee brije(x, x))$$

Jedina Erbranova instanca ove formule je:

$$(\neg brije(c, c) \vee \neg brije(c, c)) \wedge (brije(c, c) \vee brije(c, c)),$$

koja je očigledno iskazno nezadovoljiva.

Erbranova teorema – primeri

Primer

Pokažimo da je formula $\exists x.\ (P(x) \Rightarrow \forall y.\ P(y))$ *valjana.*
Primenimo metodu pobijanja i transformacije normalnih formi.

$$\neg(\exists x.\ (P(x) \Rightarrow \forall y.\ P(y)))$$

$$\forall x.\ (P(x) \wedge (\exists y.\ \neg P(y)))$$

$$\forall x.\ \exists y.\ (P(x) \wedge \neg P(y))$$

$$\forall x.\ (P(x) \wedge \neg P(f(x)))$$

Konjunkcija naredne dve instance je iskazno nezadovoljiva:

$$(P(c) \wedge \neg P(f(c))) \wedge (P(f(c)) \wedge \neg P(f(f(c))))$$

Erbranova teorema – primeri

Primer

Dokažimo da je formula $\forall x.\ \exists y.\ q(x,y)$ *logička posledica formula* $\forall x.\ \exists y.p(x,y)$ *i* $\forall x.\ \forall y.\ p(x,y) \Rightarrow q(x,y)$. *Potrebno je dokazati da je*

$$((\forall x.\ \exists y.p(x,y)) \wedge (\forall x.\ \forall y.\ p(x,y) \Rightarrow q(x,y))) \Rightarrow \forall x.\ \exists y.\ q(x,y)$$

valjana formula. NNF negacije prethodne formule je:

$$(\forall x.\ \exists y.\ p(x,y)) \wedge (\forall x.\ \forall y.\ \neg p(x,y) \vee q(x,y)) \wedge (\exists x.\ \forall y.\ \neg q(x,y))$$

Prenex transformacija daje:

$$\exists x.\ \forall z.\ \exists y.\ \forall w.\ p(z,y) \wedge (\neg p(z,w) \vee q(z,w)) \wedge \neg q(x,z)$$

Nakon skolemizacije, dobija se

$$\forall z.\ \forall w.\ p(z,f(z)) \wedge (\neg p(z,w) \vee q(z,w)) \wedge \neg q(c,z)$$

Erbranova teorema – primeri

Primer

Instance

$$p(c, f(c)) \wedge (\neg p(c, f(c)) \vee q(c, f(c))) \wedge \neg q(c, c)$$

i

$$p(f(c), f(f(c))) \wedge (\neg p(f(c), c) \vee q(f(c), c)) \wedge \neg q(c, f(c))$$

su zajedno nezadovoljive.

Primer

Primetimo, da je nezadovoljivost bilo moguće detektovati i jednostavnije da prilikom prenex transformacije nije vršena ,,ušteda" broja kvantifikatora.

$$\exists x.\ \forall x'.\ \exists y.\ \forall x''.\ \forall y'.\ \forall y''.\ p(x', y) \wedge (\neg p(x'', y') \vee q(x'', y')) \wedge \neg q(x, y''),$$

tj., nakon Skolemizacije

$$\forall x'.\ \forall x''.\ \forall y'.\ \forall y''.\ p(x', f(x')) \wedge (\neg p(x'', y') \vee q(x'', y')) \wedge \neg q(c, y'')$$

Tada bi postojala jedna jedina Erbranova instanca koja je nezadovoljiva:

$$p(c, f(c)) \wedge (\neg p(c, f(c)) \vee q(c, f(c))) \wedge \neg q(c, f(c)))$$

Erbranova teorema — zasebno instanciranje klauza

Prethodno zapažanje važi i generalno. Naime, uvođenjem novih univerzalnih kvantifikatora, moguće je postići da su slobodne promenljive u svim klauzama razdvojene (npr. $\forall x.P(x) \wedge Q(x)$ je ekvivalentno sa $\forall x'.\forall x''.P(x') \wedge P(x'')$.

Teorema

Neka je $F \equiv \forall x_1 \ldots x_n.C_1(x_1, \ldots, x_n) \wedge \ldots \wedge C_k(x_1, \ldots, x_n)$ *formula u klauzalnoj normalnoj formi. Formula je zadovoljiva ako i samo ako je skup svih baznih instanci pojedinačnih klauza iskazno zadovoljiv, tj. ako je zadovoljiv skup* $\{C_i[x_1 \to t_1] \ldots [x_n \to t_n] \mid t_1, \ldots, t_n \in H(F), 1 \leq i \leq k\}$.

Pregled

- Osnovni problem Erbran/Gilmorovog metoda dokazivanja je bio kako pronaći bazne instance koje daju nezadovoljivost.
- Faza generisanja skupova baznih instanci i faza ispitivanja njihove nezadovoljivosti su potpuno razdvojene.
- Da li je moguće nekako objediniti ove dve faze?

Primer

Posmatrajmo formulu bez kvantifikatora koja se sastoji iz klauza:

1. $\{p(x, y), q(y, f(u))\}$
2. $\{\neg q(z, z)\}$
3. $\{\neg p(w, f(w)), \neg p(f(w), f(w))\}$

(klauze su predstavljene kao skupovi literala i implicitno su univerzalno kvantifikovane). Da bismo dokazali nezadovoljivost ovog skupa klauza, razmatramo sledeće bazne instance gornjih klauza:

4.	$\{p(c, f(c)), q(f(c), f(c))\}$,	*iz* 1,	$x \to c, y \to f(c), u \to c$
5.	$\{\neg q(f(c), f(c))\}$,	*iz* 2,	$z \to f(c)$
6.	$\{\neg p(c, f(c)), \neg p(f(c), f(c))\}$,	*iz* 3,	$w \to c$
7.	$\{p(f(c), f(c)), q(f(c), f(c))\}$,	*iz* 1,	$x \to f(c), y \to f(c), u \to c$

Primenom (iskazne) rezolucije, lako se iz dobijenih baznih instanci izvodi prazna klauza (iz 4 i 5 se dobija klauza $\{p(c, f(c))\}$, iz 5 i 7 se dobija klauza $\{p(f(c), f(c))\}$. Sada klauza $\{p(c, f(c))\}$ sa klauzom 6 daje $\{\neg p(f(c), f(c))\}$, što sa klauzom $\{p(f(c), f(c))\}$ daje praznu klauzu).

Primer

(nastavak) Primetimo da je na klauzu 5 (instancu klauze 2) dva puta primenjivano pravilo rezolucije, oba puta nad (različitim) instancama klauze 1:

$$\frac{\{p(c, f(c)), q(f(c), f(c))\} \quad \{\neg q(f(c), f(c))\}}{\{p(c, f(c))\}}$$
$$\frac{\{p(f(c), f(c)), q(f(c), f(c))\} \quad \{\neg q(f(c), f(c))\}}{\{p(f(c), f(c))\}}$$

Po sličnom principu, umesto ove dve instance klauze 1 mogla je da stoji bilo koja instanca ove klauze kod koje je y zamenjeno sa $f(c)$, a x proizvoljnim baznim termom t:

$$\frac{\{p(t, f(c)), q(f(c), f(c))\} \quad \{\neg q(f(c), f(c))\}}{\{p(t, f(c))\}}$$

(rezultujuća klauza sadrži term $p(t, f(c))$). Uočeni ,,šablon" bismo mogli predstaviti na sledeći način:

$$\frac{\{p(x, f(c)), q(f(c), f(c))\} \quad \{\neg q(f(c), f(c))\}}{\{p(x, f(c))\}}$$

Primer

(nastavak) Dobijeno pravilo:

$$\frac{\{p(x, f(c)), q(f(c), f(c))\} \quad \{\neg q(f(c), f(c))\}}{\{p(x, f(c))\}}$$

predstavlja pravilo rezolucije prvog reda. Ono je na izvestan način uopštenje svih ranije opisanih primena pravila iskazne rezolucije nad instancama klauza 1 i 2, u smislu da se svaka primena pravila iskazne rezolucije nad baznim instancama klauza 1 i 2 može dobiti instanciranjem promenljive x odgovarajućim baznim termom t u gornjem pravilu. Dobijena rezolventa $\{p(x, f(c))\}$ je uopštenje svih baznih rezolventi dobijenih na ovaj način.

- *Gornje zapažanje se može formulisati i ovako: kada primenimo navedeno pravilo rezolucije prvog reda, efekat je kao da smo istovremeno primenili pravilo iskazne rezolucije nad svim baznim instancama klauza 1 i 2 gornjeg oblika.*
- *Na ovaj način, rezolucijom prvog reda se objedinjuju (ranije razdvojene) faze generisanja baznih instanci klauza i dokazivanja nezadovoljivosti.*

Primer

(nastavak) Primetimo da je za ovako nešto bilo neophodno ,,parcijalno" instancirati klauze 1 i 2 tako da se dobiju dve klauze koje sadrže suprotne literale $q(f(c), f(c))$ i $\neg q(f(c), f(c))$. Da bismo ovo postigli, bilo je potrebno pronaći uopštenu zamenu σ takvu da je $q(y, f(u))\sigma = q(z, z)\sigma$. U gornjem pravilu, korišcena je zamena $\sigma = [y \to f(c), u \to c, z \to f(c)]$. Za ovakvu zamenu kažemo da je unifikator za atome $q(y, f(u))$ i $q(z, z)$. Sada gornje pravilo rezolucije prvog reda možemo zapisati u obliku:

$$\frac{\{p(x, y), q(y, f(u))\}\sigma \quad \{\neg q(z, z)\}\sigma}{\{p(x, y)\}\sigma}$$

Definicija

Unifikator dva atoma (ili terma) P i Q je uopštena zamena σ takva da je $P\sigma = Q\sigma$. Za atome (termove) P i Q kažemo da su unifikabilni ako za njih postoji unifikator.

- Unifikator ne mora biti jedinstven.

Primer

(nastavak) Posmatrajmo ponovo klauze 1 i 2 iz našeg primera (tj. klauze $\{p(x,y), q(y,f(u))\}$ *i* $\{\neg q(z,z)\}$*). Unifikator* $\sigma = [y \to f(c), u \to c, z \to f(c)]$ *primenjen na ove dve klauze daje nam instance* $\{p(x,f(c)), q(f(c),f(c))\}$ *i* $\{\neg q(f(c),f(c))\}$*, iz kojih se pravilom rezolucije:*

$$\frac{\{p(x,f(c)), q(f(c),f(c))\} \quad \{\neg q(f(c),f(c))\}}{\{p(x,f(c))\}}$$

dobija rezolventa $\{p(x,f(c))\}$*. Sa druge strane, da smo umesto unifikatora* σ *koristili unifikator* $\tau = [y \to f(f(c)), u \to f(c), z \to f(f(c))]$*, dobili bismo instance* $\{p(x,f(f(c))), q(f(f(c)),f(f(c)))\}$ *i* $\{\neg q(f(f(c)),f(f(c)))\}$*, kao i pravilo:*

$$\frac{\{p(x,f(f(c))), q(f(f(c)),f(f(c)))\} \quad \{\neg q(f(f(c)),f(f(c)))\}}{\{p(x,f(f(c)))\}}$$

iz koga se dobija rezolventa $\{p(x,f(f(c)))\}$*.*

- Dobili smo, dakle, sledea dva pravila:

$$\frac{\{p(x, f(c)), q(f(c), f(c))\} \quad \{\neg q(f(c), f(c))\}}{\{p(x, f(c))\}}$$

$$\frac{\{p(x, f(f(c))), q(f(f(c)), f(f(c)))\} \quad \{\neg q(f(f(c)), f(f(c)))\}}{\{p(x, f(f(c)))\}}$$

- Dva dobijena pravila ,,pokrivaju" različite skupove baznih instanci klauza 1 i 2.
- Da li je moguće naći opštije pravilo rezolucije prvog reda koje pokriva sve moguće primene pravila iskazne rezolucije nad baznim instancama klauza 1 i 2 po baznim instancama literala $q(y, f(u))$, odnosno $\neg q(z, z)$?

Definicija

Kompozicija uopštenih zamena σ i τ je uopštena zamena $\sigma\tau$ definisana na sledeći način: $E(\sigma\tau) = (E\sigma)\tau$, za svaki izraz (atom ili term) E.

Definicija

Za unifikator σ atoma (termova) P i Q kažemo da je najopštiji ako je svaki drugi unifikator τ ovih atoma (termova) oblika $\tau = \sigma\tau'$, tj. τ je kompozicija unifikatora σ i neke zamene τ'.

Primetimo da za unifikator τ važi $P\tau = Q\tau$, tj. $(P\sigma)\tau' = (Q\sigma)\tau'$. Kako je $P\sigma = Q\sigma$, ovo znači da je zajednička instanca od P i Q dobijena unifikatorom τ ujedno i instanca zajedničke instance od P i Q dobijene unifikatorom σ. Dakle, najopštiji unifikator nam daje najopštiju zajedničku instancu datih izraza P i Q.

Teorema

Za svaka dva unifikabilna atoma (ili terma) P i Q postoji najopštiji unifikator koji je jedinstven do na preimenovanje promenljivih.

Primer

(nastavak) Za atome $q(y, f(u))$ *i* $q(z, z)$ *najopštiji unifikator je* $\sigma = [y \to f(u), z \to f(u)]$. *Ovim se oba atoma prezapisuju u* $q(f(u), f(u))$ *što je najopštija zajednička instanca ovih atoma. Sada najopštije pravilo rezolucije prvog reda za klauze 1 i 2 glasi:*

$$\frac{\{p(x, f(u)), q(f(u), f(u))\} \quad \{\neg q(f(u), f(u))\}}{\{p(x, f(u))\}}$$

Ovo pravilo pokriva sve moguće primene pravila iskazne rezolucije nad baznim instancama klauza 1 i 2 po baznim instancama literala $q(z, f(u)$ *i* $\neg q(z, z))$, *kao i sve moguće rezolvente koje se na taj način mogu dobiti (svaka tako dobijena bazna rezolventa će biti instanca rezolvente* $\{p(x, f(u)\})$.

Zaključujemo:

- Umesto da generišemo bazne instance i nad njima sprovodimo iskaznu rezoluciju, možemo primenjivati pravilo rezolucije prvog reda nad klauzama prvog reda na koje je primenjen najopštiji unifikator za odgovarajuće literale po kojima se rezolucija vrši
- Ostaje otvoreno pitanje: kako pronaći najopštiji unifikator?

Najopštiji unifikator: ideja algoritma

- Želimo da pronađemo najopštiji unifikator za dva atoma (ili terma) P i Q
- Ukoliko su vodeći simboli izraza P i Q različiti, jasno je da se oni ne mogu unifikovati (ovo zovemo kolizija)
- Ako su im vodeći simboli jednaki (npr. $P = r(t_1, \ldots, t_n)$ i $Q = r(s_1, \ldots, s_n)$), tada je potrebno istovremeno unifikovati parove podtermova $(t_1, s_1), \ldots, (t_n, s_n)$ (tj. pronaći zamenu σ takvu da istovremeno važi $t_1\sigma = s_1\sigma, t_2\sigma = s_2\sigma, \ldots, t_n\sigma = s_n\sigma$). Ovaj korak zovemo dekompozicija.
- Zbog toga su algoritmi za pronalaženje najopštijeg unifikatora obično i konstruisani tako da razmatraju više parova termova instovremeno na ulazu
- Dekompozicija nas u krajnjoj instanci dovodi ili do kolizije, ili do situacije u kojoj je jedan od termova u paru promenljiva. Tada je jasno da se ta promenljiva mora instancirati drugim termom u paru (ovo zovemo aplikacija)
- Problem: ako imamo par (x, t), pri čemu term t sadrži promenljivu x, tada se ta dva terma ne mogu unifikovati (ovo zovemo cikličnost). Na primer, za par $(x, f(x))$, čime god da zamenimo x imaćemo jedno f ,,viška" u desnom termu.

Algoritam za najopštiji unifikator

- Ulaz algoritma je skup parova termova $(t_1, s_1), \ldots, (t_n, s_n)$ koje treba istovremeno unifikovati
- Izlaz algoritma je skup parova $(x_1, r_1), (x_2, r_2), \ldots, (x_k, r_k)$, gde je x_i promenljiva, a r_i term koji ne sadrži x_i, ili informacija o neuspehu u slučaju neunifikabilnih termova.
- Dokle god je moguče, primenjuju se sledeći koraci (u navedenom redosledu):
  - ako postoje dva ili više identičnih parova, tada se svi osim jednog brišu (pravilo faktorisanja)
  - ako postoji par (t, t), on se uklanja iz skupa (pravilo tautologija)
  - ako postoji par (t, x), gde t nije promenljiva, tada se ovaj par zamenjuje parom (x, t) (pravilo orijentacija)
  - ako postoji par $f((t_1, \ldots, t_n), g(s_1, \ldots, s_m))$, tada se algoritam prekida, jer ne postoji unifikator (pravilo kolizija)
  - ako postoji par $(f(t_1, \ldots, t_n), f(s_1, \ldots, s_n))$, tada se taj par zamenjuje parovima $(t_1, s_1), \ldots, (t_n, s_n)$ (pravilo dekompozicija)
  - ako postoji par (x, t) pri čemu t ne sadrži promenljivu x, tada se u svim ostalim parovima x zamenjuje termom t (pravilo aplikacija)
  - ako postoji par (x, t), pri čemu t sadrži x, tada se algoritam prekida, jer ne postoji unifikator (pravilo cikličnost)
- Ukoliko algoritam nije prekinut zbog neuspeha, iz rezultujućeg skupa parova se direktno određuje najopštiji unifikator $\sigma = [x_1 \rightarrow r_1, \ldots x_k \rightarrow r_k]$.

Primer

Unifikacija atoma $q(y, f(u))$ i $q(z, z)$ iz prethodnog primera se dekompozicijom svodi na istovremenu unifikaciju parova (y, z) i $(f(u), z)$. Pravilo orijentacije nam daje skup parova $(y, z), (z, f(u))$, a pravilo aplikacije nam daje skup parova $(y, f(u)), (z, f(u))$. Dalje se ne može primeniti ni jedno od pravila, pa je rezultujući najopštiji unifikator $[y \to f(u), z \to f(u)]$.

Pregled

Metod rezolucije

- Zasniva se na ranije uvedenom pravilu rezolucije prvog reda
- U pitanju je procedura poluodlučivanja (za svaku nezadovoljivu formulu može izvođenjem prazne klauze dokazati da je nezadovoljiva)
- Pored potpunosti (za pobijanje) ima i svojstvo saglasnosti: iz zadovoljivog skupa klauza ne može se izvesti prazna klauza
- Za zadovoljiv skup klauza postoje dva scenarija: ili se više ne mogu izvoditi nove klauze pravilom rezolucije (procedura vraća odgovor da) ili se nove klauze beskonačno izvode, pa se procedura nikada ne završava

Pravilo binarne rezolucije prvog reda

Pravilo binarne rezolucije prvog reda glasi:

$$\frac{C_1 \vee A_1 \quad C_2 \vee \neg A_2}{(C_1 \vee C_2)\sigma}$$

gde je σ najopštiji unifikator atoma A_1 i A_2.

- Pravilo se zove binarno, jer uvek rezolvira dva literala (po jedan iz svake klauze).
- Da li je ovako formulisano pravilo dovoljno za dokazivanje nezadovoljivosti?
- Podsetimo se: pravilo rezolucije prvog reda bi trebalo da pokrije sve odgovarajuće primene pravila iskazne rezolucije nad baznim instancama datih klauza

Primer

Neka je dat skup klauza prvog reda („paradoks berberina"):

1. $\{\neg p(x,x), \neg p(c,x)\}$
2. $\{p(y,y), p(c,y)\}$

Ovaj skup klauza je nezadovoljiv. Zaista, ako u prvoj klauzi zamenimo x sa c, dobijamo baznu instancu $\{\neg p(c,c), \neg p(c,c)\}$, što je ekvivalentno sa $\{\neg p(c,c)\}$. Slično, zamenom y sa c u drugoj klauzi dobijamo baznu instancu $\{p(c,c)\}$. Iz ove dve bazne klauze se direktno izvodi prazna klauza.
Sa druge strane, primenom pravila rezolucije prvog reda (nad, npr. prvim literalima obe klauze) dobijamo:

$$\frac{\{\neg p(x,x), \neg p(c,x)\} \quad \{p(y,y), p(c,y)\}}{\{\neg p(c,y), p(c,y)\}}$$

pri čemu je na klauze primenjen najopštiji unifikator atoma $p(x,x)$ i $p(y,y)$ (to je zamena $\sigma = [x \to y]$). Dobijena klauza $\{\neg p(c,y), p(c,y)\}$ je ekvivalentna sa $\top$ i iz nje se dalje ništa ne može izvesti. Sličan rezultat bismo dobili za bilo koji par unifikabilnih atoma iz ovih klauza.

- Problem u prethodnom primeru je to što smo za izvođenje prazne klauze koristili bazne instance u kojima se dva različita literala prvog reda svode na isti bazni literal (npr. $p(y, y)$ i $p(c, y)$ se instanciranjem svode na bazni literal $p(c, c)$).
- Ovo dovodi do toga da bazna instanca ima manje literala od polazne klauze prvog reda
- Pravilo binarne rezolucije prvog reda ne može da ,,pokrije" iskaznu rezoluciju nad ovakvim baznim instancama
- Dva moguća rešenja ovog problema:
  - Grupisanje: parcijalno instanciranje klauze tako da se ,,slični" literali svedu na jedan, pre rezolucije
  - Uopšteno pravilo rezolucije: rezolucija koja se vrši istovremeno nad više literala iz svake od klauza

Definicija

Pravilo grupisanja ima sledeći oblik:

$$\frac{C \vee L_1 \vee L_2 \vee \ldots \vee L_m}{(C \vee L_1)\sigma}$$

gde je $C \vee L_1 \vee \ldots \vee L_m$ *klauza prvog reda takva da su literali* $L_1, \ldots, L_m$ *unifikabilni sa najopštijim unifikatorom* σ*, tj. važi da je:*

$$L_1\sigma = L_2\sigma = \ldots = L_m\sigma$$

a C *je proizvoljna disjunkcija literala prvog reda (eventualno prazna).*
NAPOMENA: primetimo da svi literali $L_1, \ldots, L_m$ *moraju biti istog polariteta (ili su svi atomi, ili su svi negacije atoma)*

- Ovo pravilo uz pravilo binarne rezolucije garantuje potpunost za pobijanje, jer omogućava izvođenje klauza koje su uopštenja baznih instanci kod kojih se više literala redukuju u jedan

Primer

(nastavak) Primenom pravila grupisanja na klauzu $\{p(y,y), p(c,y)\}$ *dobija se klauza* $\{p(c,c)\}$ *(najopštiji unifikator za literale* $p(y,y)$ *i* $p(c,y)$ *je zamena* $[y \to c]$*). Slično, primenom pravila grupisanja na klauzu* $\{\neg p(x,x), \neg p(c,x)\}$ *dobija se klauza* $\{\neg p(c,c)\}$*. Iz ove dve klauze se lako izvodi prazna klauza (binarnom rezolucijom).*

Definicija

Uopšteno pravilo rezolucije prvog reda ima sledeći oblik:

$$\frac{C' \vee A'_1 \vee A'_2 \vee \ldots \vee A'_m \quad C'' \vee \neg A''_1 \vee \neg A''_2 \vee \ldots \vee \neg A''_n}{(C' \vee C'')\sigma}$$

pri čemu su $C' \vee A'_1 \vee A'_2 \vee \ldots \vee A'_m$ *i* $C'' \vee \neg A''_1 \vee \neg A''_2 \vee \ldots \vee \neg A''_n$ *klauze prvog reda takve da su svi atomi* $A'_1, \ldots, A'_m, A''_1, \ldots, A''_n$ *unifikabilni sa najopštijim unifikatorom* σ*, tj. važi da je:*

$$A'_1\sigma = \ldots = A'_m\sigma = A''_1\sigma = \ldots = A''_n\sigma$$

a C' *i* C'' *su proizvoljne disjunkcije literala prvog reda (eventualno prazne).*

- Ovo pravilo objedinjuje pravila binarne rezolucije i grupisanja
- Njegov efekat je isti kao da smo prvo grupisali literale $A'_1, \ldots, A'_m$ u prvoj klauzi i $\neg A''_1, \ldots, \neg A''_n$ u drugoj klauzi, a zatim primenili binarnu rezoluciju nad dobijenim literalima
- Ovo pravilo je dovoljno (samo za sebe) da obezbedi potpunost (za pobijanje)

Primer

(nastavak) Primenom uopštenog pravila rezolucije nad klauzama $\{p(y,y), p(c,y)\}$ *i* $\{\neg p(x,x), \neg p(c,x)\}$ *dobijamo praznu klauzu u jednom koraku, jer je najopštiji unifikator za atome* $p(y,y)$, $p(c,y)$, $p(x,x)$ *i* $p(c,x)$ *zamena* $\sigma = [x \to c, y \to c]$.

- Setimo se da su sve klauze implicitno univerzalno kvantifikovane, svaka za sebe
- Ovo znači da možemo smatrati da su skupovi promenljivih koje se javljaju u različitim klauzama disjunktni (ovo uvek možemo postići preimenovanjem promenljivih).
- Ova činjenica nam daje dodatnu slobodu prilikom unifikacije

Primer

Skup klauza $\{p(x)\}, \{\neg p(f(x))\}$ *je nezadovoljiv iako na prvi pogled deluje da* $p(x)$ *i* $p(f(x))$ *nisu unifikabilni. Kako su ove dve klauze nezavisno (svaka za sebe) univerzalno kvantifikovane, možemo npr. u prvoj preimenovati promenljivu* x *u* x'*. Sada su atomi* $p(x')$ *i* $p(f(x))$ *unifikabilni (unifikator je* $[x' \to f(x)]$*, odakle izvodimo praznu klauzu.*

- PREPORUKA: pre primene pravila rezolucije uvek obezbediti da skupovi promenljivih u dvema odabranim klauzama budu disjunktni (preimenovanjem)

Primer

Vratimo se ponovo na primer sa početka:

1. $\{p(x, y), q(y, f(u))\}$
2. $\{\neg q(z, z)\}$
3. $\{\neg p(w, f(w)), \neg p(f(w), f(w))\}$

i dokažimo metodom rezolucije prvog reda da je ovaj skup klauza nezadovoljiv. Najpre primenjujemo pravilo rezolucije nad klauzama 1 *i* 2 *(najopštiji unifikator za atome* $q(y, f(u))$ *i* $q(z, z)$ *je* $[y \to f(u), z \to f(u)]$*). Dobijamo klauzu:*

4. $\{p(x, f(u))\}$

Dalje, primenjujemo pravilo rezolucije nad klauzama 3 *i* 4 *za prvi literal klauze* 3 *(najopštiji unifikator za atome* $p(w, f(w))$ *i* $p(x, f(u))$ *je* $[x \to w, u \to w]$*). Dobijamo rezolventu:*

5. $\{\neg p(f(w), f(w))\}$

Najzad, primenom rezolucije nad klauzama 4 *i* 5 *(unifikator* $[x \to f(w), u \to w]$*) dobijamo praznu klauzu.*

Primer

Neka je dat skup klauza:

1. $\{p(x,y), q(x,y)\}$
2. $\{\neg p(x,y), \neg q(x,y)\}$

Ovaj skup klauza je zadovoljiv, jer se iz njega ne može nikako izvesti prazna klauza (zašto?). Međutim, česta greška je da se pogrešno primeni pravilo rezolucije tako što se istovremeno ,,poništе" dva para literala: literal $p(x,y)$ sa $\neg p(x,y)$, a literal $q(x,y)$ sa $\neg q(x,y)$, i na taj način dobiju praznu klauzu. Ovako nešto nije bilo moguće ni u iskaznoj rezoluciji (npr. iz klauza $\{p,q\}$ i $\{\neg p, \neg q\}$ nije moguće izvesti praznu klauzu). Dakle, svaka primena pravila rezolucije se uvek primenjuje nad samo jednim parom literala.
Zabuna (verovatno) nastaje zbog pogrešnog razumevanja uopštenog pravila rezolucije: i ovo pravilo se uvek primenjuje nad samo jednim parom literala, s tim što se prethodno više ,,sličnih" (unifikabilnih) literala u svakoj od klauza mogu grupisati u jednu zajedničku instancu, pre primene rezolucije. Gornji primer je nešto sasvim drugo i to ne treba mešati.

Pregled

Dedukcija

- Kao i u iskaznoj logici, u logici prvog reda možemo razmatrati različite sisteme za dedukciju
- U logici prvog reda je značaj deduktivnih sistema još veći, imajući u vidu beskonačnost domena koje razmatramo
- Neki sistemi (poput prirodne dedukcije) podrazumevaju dokaze koji su čitljivi za čoveka, jer prate uobičajen način rezonovanja prilikom dokazivanja teorema (za razliku od npr. rezolucijskih dokaza nezadovoljivosti)
- Hilbertov sistem, Prirodna dedukcija, Račun sekvenata ...

Hilbertov sistem u iskaznoj logici (podsetnik)

Definicija

U iskaznoj logici, Hilbertov sistem je podrazumevao sledeće sheme aksioma:

(A1): $A \Rightarrow (B \Rightarrow A)$

(A2): $(A \Rightarrow (B \Rightarrow C)) \Rightarrow ((A \Rightarrow B) \Rightarrow (A \Rightarrow C))$

(A3): $(\neg B \Rightarrow \neg A) \Rightarrow (A \Rightarrow B)$

Kao i jedno pravilo izvođenja — modus ponens:

$$\frac{A \quad A \Rightarrow B}{B}$$

Hilbertov sistem u logici prvog reda

Definicija

U logici prvog reda, Hilbertov sistem dodatno ima i sledeće sheme aksioma:

(A4): $(\forall x)A \Rightarrow A[x \to t]$

(A5): $(\forall x)(A \Rightarrow B) \Rightarrow (A \Rightarrow (\forall x)B)$

pri čemu u aksiomi A4 term t ne sme sadržati ni jednu promenljivu koja bi zamenom x sa t postala vezana u formuli A, dok u aksiomi A5 formula A ne sme sadržati slobodna pojavljivanja promenljive x. Pored modus ponensa, dodatno pravilo izvođenja je pravilo generalizacije:

$$\frac{A}{(\forall x)A}$$

Napomene

- Kao i ranije, dokaz u Hilbertovom sistemu je niz formula, pri čemu je svaka formula nešto od sledećeg:
  - instanca aksiome
  - pretpostavka (ako je u pitanju dokaz iz nekih pretpostavki)
  - dobijena iz formula koje joj prethode u nizu primenom pravila modus ponens ili pravila generalizacije
- Kao i ranije, veznici $\wedge$, $\vee$ se definišu kao $\neg(A \Rightarrow \neg B)$, odnosno $(\neg A) \Rightarrow B$
- Egzistencijalni kvantifikator $(\exists x)A$ se definiše kao $\neg(\forall x)\neg A$

Primer

Dokažimo da važi:

$$A, (\forall x)A \Rightarrow C \vdash (\forall x)C$$

Dokaz je sledeći niz formula:

1.	A	(Hip)
2.	$(\forall x)A$	$(1, Gen)$
3.	$(\forall x)A \Rightarrow C$	(Hip)
4.	C	$(2, 3, MP)$
5.	$(\forall x)C$	$(4, Gen)$

Još napomena

- Setimo se da bi deduktivni sistem trebalo da bude saglasan, tj. da se u njemu mogu dokazati samo valjane formule
- Da bi to bilo ispunjeno, potrebno je da sve aksiome budu valjane, kao i da se pravilima izvođenja iz valjanih formula izvode valjane formule
- Aksiomatske sheme A1, A2 i A3, kao i pravilo modus ponens zadovoljavaju ove uslove (to znamo iz iskazne logike)
- Pravilo generalizacije takođe zadovoljava ovaj uslov (zašto?)
- Što se tiče aksiomatskih shema A4 i A5, one imaju dodatne uslove upravo zato da bismo obezbedili da sve njihove instance budu valjane formule
- Na primer, navedeno ograničenje vezano za term t u aksiomi A4 sprečava da imamo npr. instancu $(\forall x)(\exists y)p(x,y) \Rightarrow (\exists y)p(y,y)$ koja nije valjana (uzmimo npr. strukturu celih brojeva u kojoj predikat $p(x,y)$ ima značenje „y je sledbenik broja x")
- Slično, ograničenje navedeno u aksiomi A5 sprečava da imamo instancu $(\forall x)(p(x) \Rightarrow p(x)) \Rightarrow (p(x) \Rightarrow (\forall x)p(x))$ koja nije valjana (neka npr. $p(x)$ znači „x je parno" u skupu celih brojeva)

Teorema

(Teorema o dedukciji) Ako važi $\Gamma, A \vdash B$ *i ako za to tvrđenje postoji dokaz u kome se ne primenjuje pravilo generalizacije ni po jednoj promenljivoj slobodnoj u* A*, tada važi* $\Gamma \vdash A \Rightarrow B$

- Gornje ograničenje je posledica ograničenja iz aksiomatske sheme A5 koja se koristi u (meta)dokazu teoreme o dedukciji
- Teorema o dedukciji svakako važi ako je formula A rečenica

Teorema

(Obrat teoreme o dedukciji) Ako važi $\Gamma \vdash A \Rightarrow B$*, tada važi* $\Gamma, A \vdash B$*.*

- Obrat teoreme o dedukciji važi bez ograničenja i dodatnih uslova

Primer

Posmatrajmo tvrđenje $p(x) \vdash (\forall x)p(x)$. *Ovo tvrđenje važi u Hilbertovom sistemu (dokaz se svodi na jednu primenu pravila generalizacije). Međutim, tvrđenje* $\vdash p(x) \Rightarrow (\forall x)p(x)$ *nije teorema Hilbertovog sistema, jer bismo u dokazu tog tvrđenja morali da koristimo instancu aksiomatske sheme A5* $(\forall x)(p(x) \Rightarrow p(x)) \Rightarrow (p(x) \Rightarrow (\forall x)p(x))$ *za koju smo već ranije konstatovali da nije dozvoljena (jer nije valjana). Zaista, formula* $p(x) \Rightarrow (\forall x)p(x)$ *nije valjana, pa je ne možemo dokazati ni u jednom saglasnom deduktivnom sistemu.*

Izvedena pravila Hilbertovog sistema

Sledeća tvrđenja važe u Hilbertovom sistemu, a mogu se koristiti kao pravila za izvođenje kraćih i intuitivnijih dokaza:

Pravilo A $(\forall x)A \vdash A[x \to t]$, gde je t term koji ne sadrži ni jednu promenljivu koja bi zamenom x sa t postala vezana u A (specijalno, važi $(\forall x)A \vdash A$, zamenom $x \to x$)

Pravilo E $A[x \to t] \vdash (\exists x)A$ (uz isti uslov kao i u pravilu A)

Meta-tvrđenja o Hilbertovom sistemu

Teorema

Hilbertov sistem je saglasan, tj. u njemu je moguće dokazati samo valjane formule.

Teorema

Hilbertov sistem je potpun, tj. u njemu je moguće dokazati svaku valjanu formulu.

Prirodna dedukcija u logici prvog reda

Prirodna dedukcija

- Deduktivni sistem koji je najbliži načinu rezonovanja čoveka prilikom dokazivanja matematičkih tvrđenja
- Postoje pravila uvođenja i pravila eliminacije veznika
- Pored pravila za iskazne veznike (koja su ista kao i u prirodnoj dedukciji za iskaznu logiku) imamo pravila za kvantifikatore
- Takođe, postoje dodatna pravila za klasičnu logiku (ista kao i u iskaznom slučaju)

Pravila prirodne dedukcije za iskazne veznike (podsetnik)

Negacija

$$\dfrac{\begin{array}{c}[A]^1\\ \vdots\\ \bot\end{array}}{\neg A}\ \neg I^1 \qquad \dfrac{A \quad \neg A}{\bot}\ \neg E$$

Konjunkcija

$$\dfrac{A \quad B}{A \wedge B}\ \wedge I \qquad \dfrac{A \wedge B}{A}\ \wedge E1 \qquad \dfrac{A \wedge B}{B}\ \wedge E2$$

Disjunkcija

$$\dfrac{A}{A \vee B}\ \vee I1 \qquad \dfrac{B}{A \vee B}\ \vee I2 \qquad \dfrac{A \vee B \quad \begin{array}{c}[A]^1\\ \vdots\\ C\end{array} \quad \begin{array}{c}[B]^2\\ \vdots\\ C\end{array}}{C}\ \vee E^{1,2}$$

Pravila prirodne dedukcije za iskazne veznike (podsetnik)

Implikacija

$$\frac{\begin{array}{c}[A]^1\\ \vdots\\ B\end{array}}{A \Rightarrow B} \Rightarrow I^1 \qquad \frac{A \qquad A \Rightarrow B}{B} \Rightarrow E$$

Logičke konstante

$$\frac{\bot}{A} \bot E \qquad \frac{}{\top} \top I$$

Klasična naspram intuicionističke logike (podsetnik)

Klasična pravila

$$\frac{}{A \vee \neg A}\ \text{ExcludedMiddle} \qquad \frac{\neg\neg A}{A}\ \text{DoubleNegation} \qquad \frac{\begin{array}{c}[\neg A]^1 \\ \vdots \\ \bot\end{array}}{A}\ \textit{Contradiction}^1$$

Pravila za kvantifikatore

Univerzalni kvantifikator

$$\frac{A[x \to y]}{(\forall x)A}\ \forall I \qquad \frac{(\forall x)A}{A[x \to t]}\ \forall E$$

Egzistencijalni kvantifikator

$$\frac{A[x \to t]}{(\exists x)A}\ \exists I \qquad \frac{(\exists x)A \qquad \begin{array}{c}[A[x \to y]]^1 \\ \vdots \\ B\end{array}}{B}\ \exists E^1$$

Dokazi u prirodnoj dedukciji

Dokaz u prirodnoj dedukciji je stablo u čijim se čvorovima nalaze formule. U korenom čvoru se nalazi formula koja se dokazuje, u listovima se nalaze aksiome ili pretpostavke (od kojih neke mogu biti oslobođene primenom određenih pravila), a svaki čvor v koji nije list sadrži formulu koja se dobija od formula u čvorovima koji su deca čvora v, primenom nekog od pravila. Formula je teorema ako za nju postoji dokaz bez neoslobođenih pretpostavki.

Uvođenje univerzalnog kvantifikatora

Uvođenje univerzalnog kvantifikatora

Pravilo za uvođenje univerzalnog kvantifikatora glasi:

$$\frac{A[x \to y]}{(\forall x)A}\ \forall I$$

uz dodatni uslov da se promenljiva y ne nalazi kao slobodna ni u $(\forall x)A$ ni u bilo kojoj neoslobođenoj pretpostavci u stablu dokaza formule $A[x \to y]$. Intuitivno, ovo znači sledeće: ako formula A važi za neki element y o kome nemamo nikakvih pretpostavki (tj. potpuno je proizvoljan), tada A važi za svaki element x. Specijalno, ako uzmemo da je $x = y$, tada dobijamo oblik:

$$\frac{A}{(\forall x)A}\ \forall I$$

uz uslov da se promenljiva x ne nalazi slobodna u neoslobođenim pretpostavkama u dokazu formule A (tj. A važi za proizvoljan element x). Ovaj oblik pravila odgovara pravilu generalizacije u Hilbertovom sistemu. Međutim, tamo nije bilo dodatnog uslova. Sledeći primer ilustruje ovo važno pitanje.

Primer

Tvrđenje $p(x) \vdash (\forall x)p(x)$ *je važilo u Hilbertovom sistemu (jedna primena pravila generalizacije). Međutim, ovo ne važi u prirodnoj dedukciji, jer bi zahtevalo primenu pravila uvođenja univerzalnog kvantifikatora, uz neoslobođenu pretpostavku* $p(x)$ *koja sadrži slobodnu promenljivu* x.
Sa druge strane, tvrđenje $\vdash p(x) \Rightarrow (\forall x)p(x)$ *ne važi ni u Hilbertovom sistemu, ni u prirodnoj dedukciji. U Hilbertovom sistemu tvrđenje* $p(x) \vdash (\forall x)p(x)$ *važi, ali se teorema o dedukciji ne može primeniti zato što* $p(x)$ *sadrži slobodnu promenljivu po kojoj je prethodno izvršena generalizacija. U slučaju prirodne dedukcije, već tvrđenje* $p(x) \vdash (\forall x)p(x)$ *ne važi. Ako bi ono važilo, tada bismo lako, pravilom uvođenja implikacije, izveli formulu* $p(x) \Rightarrow (\forall x)p(x)$, *uz oslobađanje pretpostavke* $p(x)$.
Dakle, u prirodnoj dedukciji se izvođenje ne-valjanih formula poput $p(x) \Rightarrow (\forall x)p(x)$ *sprečava ograničavanjem uvođenja univerzalnih kvantifikatora (dok je kod Hilbertovog sistema generalizacija potpuno slobodna), dok se kod Hilbertovog sistema to sprečava ograničenjem koje postoji u teoremi o dedukciji, tj. u aksiomi A5 (dok je kod prirodne dedukcije uvođenje implikacije potpuno slobodno). U oba slučaja, krajnji efekat je isti.*

Eliminacija univerzalnog kvantifikatora

Pravilo za eliminaciju univerzalnog kvantifikatora glasi:

$$\frac{(\forall x)A}{A[x \to t]}\ \forall E$$

Ovo pravilo je prilično intuitivno: ako A važi za svako x, tada to x možemo zameniti bilo kojim termom t (tj. važi proizvoljna instanca (po x) formule A). Ovo pravilo odgovara izvedenom pravilu A u Hilbertovom sistemu, a ograničenje za njegovu primenu je slično kao i kod izvedenog pravila A: term t ne sme sadržati promenljive koje bi zamenom x sa t u A postale vezane.

Primer

Pođimo od pretpostavke $(\forall x)(\exists y)p(x,y)$. *Ako ne bi bilo ograničenja u primeni pravila eliminacije univerzalnog kvantifikatora, tada bismo mogli da njegovom primenom iz ove pretpostavke izvedemo formulu* $(\exists y)p(y,y)$. *Dalje bismo mogli primenom pravila uvođenja implikacije da izvedemo formulu* $(\forall x)(\exists y)p(x,y) \Rightarrow (\exists y)p(y,y)$ *(uz oslobađanje pretpostavke). Za ovu formulu smo ranije konstatovali da nije valjana. Ovaj primer ilustruje neophodnost uvedenog ograničenja prilikom primene pravila eliminacije univerzalnog kvantifikatora.*

Uvođenje egzistencijalnog kvantifikatora

Pravilo za uvođenje egzistencijalnog kvantifikatora glasi:

$$\frac{A[x \to t]}{(\exists x)A}\ \exists I$$

Ovo pravilo je takođe prilično intuitivno: ako važi neka instanca (po x) formule A, tj. važi $A[x \to t]$ za neko t, tada znači da postoji neka vrednost promenljive x za koju je formula A tačna. Otuda, važi $(\exists x)A$. Ovo pravilo odgovara izvedenom pravilu E u Hilbertovom sistemu, a ograničenje za njegovu primenu je slično kao i kod pravila E: term t ne sme sadržati promenljive koje bi postale vezane zamenom x sa t u A.

Primer

Pođimo od pretpostavke $(\forall y)p(y, y)$. *Kada ne bi bilo dodatnog uslova u primeni pravila za uvođenje egzistencijalnog kvantifikatora, mogli bismo da njegovom primenom izvedemo npr. formulu* $(\exists x)(\forall y)p(x, y)$. *Sada bismo primenom pravila uvođenja implikacije, uz eliminaciju pretpostavke, dobili formulu* $(\forall y)p(y, y) \Rightarrow (\exists x)(\forall y)p(x, y)$. *Ova formula nije valjana (uzmimo proizvoljnu strukturu sa bar dva elementa i interpretirajmo* $p(x, y)$ *kao ,,x je jednako y").*

Eliminacija egzistencijalnog kvantifikatora

Pravilo eliminacije egzistencijalnog kvantifikatora glasi:

$$\frac{(\exists x)A \qquad \begin{array}{c}[A[x \to y]]^1 \\ \vdots \\ B\end{array}}{B}\ \exists E^1$$

uz dodatni uslov da se promenljiva y ne pojavljuje kao slobodna ni u formuli B, ni u formuli $(\exists x)A$, ni u neoslobođenim pretpostavkama u dokazu formule B, osim možda u samoj formulu $A[x \to y]$. Intuitivno, ovo znači sledeće: ako pretpostavimo da formula A važi za neki proizvoljan element y (tj. za element za koji nema nikakvih dodatnih pretpostavki), i ako iz te pretpostavke možemo da dokažemo formulu B, to znači da iz $(\exists x)A$ možemo da dokažemo B.

Primer

Sledeći primer ilustruje neophodnost uvođenja dodatnih uslova u pravilo eliminacije egzistencijalnog kvantifikatora: pretpostavimo da imamo pretpostavku $(\exists x)p(x)$. *Kada ne bi bilo dodatnih uslova u pravilu za eliminaciju egzistencijalnog kvantifikatora, tada bismo mogli da (stavljanjem* $A = B = p(x)$ *i* $y = x$*) ovim pravilom izvedemo* $p(x)$. *Dalje bismo mogli da, primenom pravila uvođenja univerzalnog kvantifikatora, izvedemo formulu* $(\forall x)p(x)$ *(jer jedina neoslobođena pretpostavka u dokazu formule* $p(x)$*, tj. formula* $(\exists x)p(x)$ *ne sadrži slobodna pojavljivanja promenljive* x*). Dalje se primenom pravila uvođenja implikacije oslobađa pretpostavka* $(\exists x)p(x)$ *i izvodi se formula* $(\exists x)A \Rightarrow (\forall x)A$*, koja naravno ne bi trebalo da bude dokaziva, jer nije valjana. Dodatni uslov u pravilu eliminacije egzistencijalnog kvantifikatora sprečava ovakva izvođenja.*

Meta-tvrđenja o prirodnoj dedukciji

Teorema

Sistem prirodne dedukcije za logiku prvog reda je saglasan, tj. u njemu je moguće dokazati samo valjane formule prvog reda.

Teorema

Sistem prirodne dedukcije za logiku prvog reda je potpun, tj. u njemu je moguće dokazati svaku valjanu formulu prvog reda.

Primer

Dokažimo da važi $\vdash (\exists x)(\forall y)p(x,y) \Rightarrow (\forall y)(\exists x)p(x,y)$. *Dokaz se može predstaviti sledećim stablom:*

$$\dfrac{\dfrac{[(\exists x)(\forall y)p(x,y)]^{\,2} \qquad \dfrac{\dfrac{\dfrac{[(\forall y)p(x',y)]^{\,1}}{p(x',y')}\forall E}{(\exists x)p(x,y')}\exists I}{(\forall y)(\exists x)p(x,y)}\forall I}{(\forall y)(\exists x)p(x,y)}\exists E,1}{(\exists x)(\forall y)p(x,y) \Rightarrow (\forall y)(\exists x)p(x,y)} \Rightarrow I,2$$

Intuitivno, dokaz teče na sledeći način: da bismo dokazali tvrđenje, treba da dokažemo da se formula $(\forall y)(\exists x)p(x,y)$ *može dokazati iz* $(\exists x)(\forall y)p(x,y)$ *(nakon toga je dovoljno primeniti pravilo uvođenja implikacije). Da bismo dokazali* $(\forall y)(\exists x)p(x,y)$, *potrebno je da dokažemo da* $(\exists x)p(x,y')$ *za važi za proizvoljno* y'. *Pošto smo pretpostavili* $(\exists x)(\forall y)p(x,y)$, *znamo da važi* $(\forall y)p(x',y)$ *za neko proizvoljno* x'. *Međutim, zbog univerzalnog kvantifikatora tada važi* $p(x',y')$ *za to izabrano* x' *i* y', *odakle važi* $(\exists x)p(x,y')$ *za naše proizvoljno* y', *što je i trebalo dokazati.*

Primer

Dokažimo sledeću formulu: $\neg(\exists x)p(x) \Rightarrow (\forall y)\neg p(y)$. *Dokaz se može predstaviti sledećim stablom:*

$$\dfrac{\dfrac{\dfrac{[p(z)]^{\,1}}{(\exists x)p(x)}\,\exists I \qquad [\neg(\exists x)p(x)]^{\,2}}{\dfrac{\bot}{\neg p(z)}\,\neg I, 1}\,\neg E}{\dfrac{(\forall y)\neg p(y)}{\neg(\exists x)p(x) \Rightarrow (\forall y)\neg p(y)}\,\forall I}\,\Rightarrow I, 2$$

Intuitivno: potrebno je dokazati formulu $(\forall y)\neg p(y)$ *iz pretpostavke* $\neg(\exists x)p(x)$. *Neka* $p(z)$ *važi za neko proizvoljno* z. *Tada važi tvrđenje* $(\exists x)p(x)$, *što je u kontradikciji sa pretpostavkom* $\neg(\exists x)p(x)$. *Ovo znači da nije* $p(z)$, *tj. da važi* $\neg p(z)$. *Kako je* z *proizvoljno, tj. za njega ne važe nikakve dodatne pretpostavke, možemo da zaključimo da važi* $(\forall y)\neg p(y)$. *Otuda sledi tvrđenje.*

- U sistemu Isabelle postoje pravila za eliminaciju i uvođenje univerzalnog i egzistencijalnog kvantifikatora.

$$
\begin{array}{rcl}
\mathit{allI} & : & \bigwedge x.\ P\ x \Longrightarrow (\forall\ x.\ P\ x) \\
\mathit{allE} & : & [\![\forall\ x.\ P\ x;\ P\ ?x \Longrightarrow R]\!] \Longrightarrow R \\
\mathit{exI} & : & P\ ?x \Longrightarrow (\exists\ x.\ P\ x) \\
\mathit{exE} & : & [\![\exists\ x.\ P\ x;\ \bigwedge x.\ P\ x \Longrightarrow R]\!] \Longrightarrow R
\end{array}
$$

$\bigwedge$ označava proizvoljnu promenljivu o kojoj nema nikakvih dodatnih pretpostavki, dok $?x$ označava proizvoljni term (uslovi vezivanja promenljivih se rešavaju automatskim preimenovanjem tokom supstitucije).

Primer dokaza u sistemu Isabelle

```
lemma "(Ex x. All y. P x y) --> (All y. Ex x. P x y)"
apply (rule impI)
apply (rule allI)
apply (erule exE)
apply (erule_tac x="y" in allE)
apply (rule_tac x="x" in exI)
apply assumption
done
```