

Automatsko rezonovanje – beleške sa predavanja Rezonovanje u logici prvog reda

Milan Banković (po slajdovima Filipa Marića)

* Matematički fakultet,
Univerzitet u Beogradu

Prolećni semestar 2026/27.

Pregled

- 1 Metod tabloa u logici prvog reda
- 2 Metode zasnovane na Erbranovoj teoremi
- 3 Metod rezolucije u logici prvog reda

Metod tabloa

Metod tabloa

- Metod tabloa za iskaznu logiku se može proširiti i na logiku prvog reda
- Pravila za iskazne veznike ostaju ista kao i ranije
- Uvode se nova pravila za tretman kvantifikatora
- Nezadovoljivost se dokazuje konstrukcijom tabloa čije su sve grane zatvorene
- Valjanost se dokazuje tako što se dokaže da je negacija nezadovoljiva

Iskazni tablo – podsetnik

Pravila za iskazne veznike

$$\frac{\neg\neg A}{A}$$

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

$$B$$

$$\frac{\neg(A \vee B)}{\neg A}$$

$$\neg B$$

$$\frac{\neg(A \Rightarrow B)}{A}$$

$$\neg B$$

$$\frac{\neg(A \wedge B)}{\neg A \mid \neg B}$$

$$\frac{A \vee B}{A \mid B}$$

$$\frac{A \Rightarrow B}{\neg A \mid B}$$

Pravila za kvantifikatore

Uvodimo dodatna pravila za kvantifikatore:

$$\frac{\forall x.A}{A[x \rightarrow t]}$$

$$\frac{\neg \exists x.A}{\neg A[x \rightarrow t]}$$

$$\frac{\exists x.A}{A[x \rightarrow a]}$$

$$\frac{\neg \forall x.A}{\neg A[x \rightarrow a]}$$

U gornjim pravilima:

- t je bilo koji bazni term
- a je **novi** simbol konstante
- Leva pravila se mogu primenjivati više puta za istu formulu (ali za različite termove t)
- Desna pravila se za svaku formulu tog tipa primenjuju samo jednom

Metod tabloa

Primer

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

|

$$\neg(P(a) \Rightarrow (\forall y.P(y)))$$

|

$$P(a)$$

|

$$\neg\forall y.P(y)$$

|

$$\neg P(b)$$

|

$$\neg(P(b) \Rightarrow (\forall y.P(y)))$$

|

$$P(b)$$

|

$$\neg\forall y.P(y)$$

Metod tabloa

Primer

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

$$\begin{array}{c} | \\ \exists x.\forall y.P(x,y) \end{array}$$

$$\begin{array}{c} | \\ \neg(\forall y.\exists x.P(x,y)) \end{array}$$

$$\begin{array}{c} | \\ \forall y.P(a,y) \end{array}$$

$$\begin{array}{c} | \\ \neg(\exists x.P(x,b)) \end{array}$$

$$\begin{array}{c} | \\ P(a,b) \end{array}$$

$$\begin{array}{c} | \\ \neg P(a,b) \end{array}$$

Metod tabloa

Metod poluodlučivanja

Može se pokazati:

- ako je formula u korenu tabloa **nezadovoljiva**, tada se uvek može, sistematskom primenom pravila, konstruisati zatvoren tablo, čime se algoritam zaustavlja
- ako je formula u korenu tabloa **zadovoljiva**, tada nije moguće konstruisati zatvoren tablo; štaviše, tablo može imati beskonačne grane, pa se algoritam u opštem slučaju ne zaustavlja

Otuda je metod tabloa samo metod **poluodlučivanja** za problem **nezadovoljivosti** formula prvog reda u proizvoljnom obliku:

- tipično se koristi kao metod poluodlučivanja za dualni problem **valjanosti**

Pregled

- 1 Metod tabloa u logici prvog reda
- 2 Metode zasnovane na Erbranovoj teoremi
- 3 Metod rezolucije u logici prvog reda

Mehanizacija Erbranove teoreme

Kako iskoristiti Erbranovu teoremu u praksi?

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

Teorema

Univerzalno kvantifikovana formula je nezadovoljiva akko postoji konačan nezadovoljiv podskup njenih baznih instanci.

Mehanizacija Erbranove teoreme

Procedure poluodlučivanja zasnovane na Erbranovoj teoremi

- Procedure poluodlučivanja Erbranovog tipa nabrajaju skupove baznih instanci dodajući nove bazne instance u tekući skup:
 - Ukoliko se utvrdi nezadovoljivost tekućeg skupa instanci, polazna formula je nezadovoljiva
 - Ukoliko se iscrpe sve bazne instance, a nezadovoljivost se ne utvrdi, polazna formula je zadovoljiva
 - S obzirom na kompaktnost, u slučaju nezadovoljive formule ćemo sigurno u konačnom broju koraka doći do konačnog nezadovoljivog podskupa, pod uslovom da se bazne instance nabrajaju sistematski
 - U slučaju da je formula zadovoljiva, takvog konačnog nezadovoljivog podskupa nema, pa će u slučaju beskonačnog Erbranovog univerzuma procedura beskonačno dodavati nove bazne instance i neće se zaustaviti
- Ispitivanje zadovoljivosti konačnog skupa baznih instanci se svodi iskaznom apstrakcijom na problem iskazne zadovoljivosti
- Najčuenija je [Gilmorova](#) procedura (prva implementirana) koja instance prevodi u DNF i tako ispituje njihovu zadovoljivost
- Moderniji pristup može podrazumevati upotrebu CDCL SAT rešavača za ispitivanje zadovoljivosti skupova baznih instanci (transformisanih u KNF)

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 transformacija daje njoj ekvivalentnu formulu $\exists z. \exists u. \forall w. P(z, w) \wedge \neg P(w, u)$. Skolemizacijom se dobija formula $\forall w. P(c_1, w) \wedge \neg P(w, c_2)$ koja je nezadovoljiva 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:

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

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

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

Jedina Erbranova instanca ove formule je:

$$(\neg \text{brije}(c, c) \vee \neg \text{brije}(c, c)) \wedge (\text{brije}(c, c) \vee \text{brije}(c, c)),$$

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 nezadovoljljiva:

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

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.

Erbranova teorema – primeri

Primer

Primetimo, da je nezadovoljivost bilo moguće detektovati i jednostavnije da prilikom prenex transformacije nije vršena „ušteta” 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

Odvojena kvantifikacija 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 $(\forall x. P(x)) \wedge (\forall x. Q(x))$, odnosno sa $(\forall x'. P(x')) \wedge (\forall x''. Q(x''))$, tj. sa $\forall x'. \forall x''. P(x') \wedge Q(x'')$). Ovo nam omogućava da svaku klauzu instanciramo zasebno!

Erbranova teorema za klauzalnu formu

Teorema

Neka je $F \equiv \forall x_1 \dots x_n. C_1(x_1, \dots, x_n) \wedge \dots \wedge C_k(x_1, \dots, x_n)$ 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 $\{C_i[x_1 \rightarrow t_1, \dots, x_n \rightarrow t_n] \mid t_1, \dots, t_n \in H(F), 1 \leq i \leq k\}$.

Erbranova teorema – moderan pristup

CDCL + instanciranje klauza

Neka F skup klauza prvog reda i neka je S (inicijalno $S = \emptyset$) tekući skup baznih instanci klauza iz F . U svakoj iteraciji petlje:

- generišemo novu baznu instancu neke klauze iz F i dodajemo je u S
- CDCL SAT rešavačem ispitujemo zadovoljivost skupa S
 - ako je S nezadovoljiv, tada je i F nezadovoljiv
 - u suprotnom, prelazimo na sledeću iteraciju petlje

Napomena

CDCL SAT rešavači imaju svojstvo inkrementalnosti:

- pri dodavanju nove klauze, ne započinje se pretraga iznova
- umesto toga, ako je novododata klauza u konfliktu sa već izgrađenim modelom, pokreće se analiza konflikta i vraćamo se na nivo povratnog skoka

Erbranova teorema – moderan pristup

Gde se koristi ovaj pristup?

Moderni **SMT rešavači** koriste instanciranje za ispitivanje zadovoljivosti univerzalno kvantifikovanih formula.

Nedostaci

Glavni problem sa ovakvim metodama je kako odabrati instance koje će se generisati:

- sistematski pristup: izuzetno neefikasan!
- heuristički pristup: po pravilu se gubi kompletnost za pobijanje!
- SMT rešavači koriste heurističke pristupe

Pregled

- 1 Metod tabloa u logici prvog reda
- 2 Metode zasnovane na Erbranovoj teoremi
- 3 Metod rezolucije u logici prvog reda**

Neefikasnost metoda zasnovanih na Erbranovoj teoremi

Problem efikasnosti

- Osnovni problem metoda zasnovanih na Erbranovoj teoremi 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

Primer

Posmatrajmo univerzalno kvantifikovanu formulu 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, | $x \rightarrow c, y \rightarrow f(c), u \rightarrow c$ |
| 5. $\{\neg q(f(c), f(c))\},$ | iz 2, | $z \rightarrow f(c)$ |
| 6. $\{\neg p(c, f(c)), \neg p(f(c), f(c))\},$ | iz 3, | $w \rightarrow c$ |
| 7. $\{p(f(c), f(c)), q(f(c), f(c))\},$ | iz 1, | $x \rightarrow f(c), y \rightarrow f(c), u \rightarrow c$ |

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).

Uopštenje iskazne rezolucije

Primer

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

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

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:

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

Uopštenje iskazne rezolucije

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.

Unifikator

Primer

(nastavak) Primitimo 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, korišćena je zamena $\sigma = [y \rightarrow f(c), u \rightarrow c, z \rightarrow f(c)]$. 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.

Važna napomena

Unifikator ne mora biti jedinstven.

Primer - unifikacija

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 $\sigma = [y \rightarrow f(c), u \rightarrow c, z \rightarrow f(c)]$ 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 $\tau = [y \rightarrow f(f(c)), u \rightarrow f(c), z \rightarrow f(f(c))]$, 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 $\{p(x, f(f(c)))\}$.

- Jasno je da dobijena dva 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 $\neg q(z, z)$?

Najopštiji unifikator

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' .*

Napomena

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.

Jedinstvenost najopštijeg unifikatora

Dokaz

Dokaz postojanja najopštijeg unifikatora izostavljamo zbog kompleksnosti. Dokažimo samo jedinstvenost. Neka su σ_1 i σ_2 dva najopštija unifikatora za termine t i s . Iz činjenice da je σ_1 najopštiji unifikator sledi da se unifikator σ_2 može zapisati kao $\sigma_2 = \sigma_1\tau_1$. Slično, iz činjenice da je σ_2 najopštiji unifikator sledi da se unifikator σ_1 može zapisati kao $\sigma_1 = \sigma_2\tau_2$. Otuda je $\sigma_2 = (\sigma_2\tau_2)\tau_1 = \sigma_2(\tau_2\tau_1)$. Ovo znači da je zamena $\tau_2\tau_1$ identička zamena (koja svaku promenljivu zamenjuje samom sobom), što je moguće samo ako su τ_1 i τ_2 preimenovanja promenljivih (tj. svaku promenljivu zamenjuju nekom drugom promenljivom). Drugim rečima, najopštiji unifikator je jedinstven do na preimenovanje promenljivih.

Primer - najopštiji unifikator

Primer

(nastavak) Za atome $q(y, f(u))$ i $q(z, z)$ najopštiji unifikator je $\sigma = [y \rightarrow f(u), z \rightarrow f(u)]$. 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

Kako pronaći najopštiji unifikator?

- 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. $P = r(t_1, \dots, t_n)$ i $Q = r(s_1, \dots, s_n)$), tada je potrebno **istovremeno** unifikovati parove podtermova $(t_1, s_1), \dots, (t_n, s_n)$ (tj. pronaći zamenu σ takvu da istovremeno važi $t_1\sigma = s_1\sigma, t_2\sigma = s_2\sigma, \dots, t_n\sigma = s_n\sigma$). Ovaj korak zovemo **dekompozicija**.
- Zbog toga su algoritmi za pronalaženje najopštijeg unifikatora obično i konstruisani tako da razmatraju više parova termova istovremeno 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

Algoritam

- Ulaz algoritma je skup parova termova (ili atoma) $(t_1, s_1), \dots, (t_n, s_n)$ koje treba istovremeno unifikovati
- Izlaz algoritma je skup parova $(x_1, r_1), (x_2, r_2), \dots, (x_k, r_k)$, 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 moguće, 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 $(f(t_1, \dots, t_n), g(s_1, \dots, s_m))$, tada se algoritam prekida, jer ne postoji unifikator (pravilo **kolizija**)
 - ako postoji par $(f(t_1, \dots, t_n), f(s_1, \dots, s_n))$, tada se taj par zamenjuje parovima $(t_1, s_1), \dots, (t_n, s_n)$ (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 $\sigma = [x_1 \rightarrow r_1, \dots, x_k \rightarrow r_k]$.

Primer - najopštiji unifikator

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 $[y \rightarrow f(u), z \rightarrow f(u)]$.

Metod rezolucije

O metodu 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 (dakle, metod nema svojstvo **zaustavljanja**)

Pravilo binarne rezolucije prvog reda

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

Pravilo binarne rezolucije

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 $\sigma = [x \rightarrow y]$). 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.

Pravilo binarne rezolucije

U čemu je problem?

- Problem u prethodnom primeru je to što smo za izvođenje prazne klauze koristili bazne instance u kojima se dva različita literala prvog reda svedu na isti bazni literal (npr. $p(y, y)$ i $p(c, y)$ se instanciranjem svedu 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

Pravilo grupisanja

Definicija

Pravilo grupisanja ima sledeći oblik:

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

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

$$L_1\sigma = L_2\sigma = \dots = L_m\sigma$$

a C je proizvoljna disjunkcija literala prvog reda (eventualno prazna).

NAPOMENA: primetimo da svi literali $L_1, \dots, L_m$ moraju biti istog polariteta (ili su svi atomi, ili su svi negacije atoma)

Napomena

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

Pravilo grupisanja

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 $[y \rightarrow c]$). 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).

Uopšteno pravilo rezolucije

Definicija

Uopšteno pravilo rezolucije prvog reda ima sledeći oblik:

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

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

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

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

Napomene

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

Uopšteno pravilo rezolucije

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 $\sigma = [x \rightarrow c, y \rightarrow c]$.

Preimenovanje promenljivih u klauzama

Podsetnik – napomena

- 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 disjunktne (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 $[x' \rightarrow f(x)]$), odakle izvodimo praznu klauzu.

PREPORUKA

Pre primene pravila rezolucije uvek obezbediti da skupovi promenljivih u dvema odabranim klauzama budu disjunktne (preimenovanjem)

Metod rezolucije

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 $[y \rightarrow f(u), z \rightarrow f(u)]$). Dobijamo klauzu:

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

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 $[x \rightarrow w, u \rightarrow w]$). Dobijamo rezolventu:

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

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

Metod rezolucije

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, studenti često pogrešno primene pravilo rezolucije tako što istovremeno „ponište” 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.