

Automatsko rezonovanje – beleške sa predavanja Teorije prvog reda i SMT

Milan Banković

*Matematički fakultet,
Univerzitet u Beogradu

Prolećni semestar 2023/24.

Pregled

- 1 Teorije prvog reda
- 2 SMT problem i SMT rešavači

Uvod i motivacija

Logika prvog reda i jednakosna logika

- U opštoj logici prvog reda smo sve simbole iz signature mogli da interpretiramo na proizvoljan način
- U logici sa jednakošću smo imali predikatski simbol jednakosti čija je interpretacija bila fiksirana, dok su se ostali simboli mogli interpretirati proizvoljno
- Simbol jednakosti se uvek interpretirao kao jednakost na odabranom domenu, tj. razmatrali smo samo **normalne** interpretacije
- Drugim rečima, jednakosna logika se može razumeti kao semantičko suženje logike prvog reda, jer ograničavamo skup interpretacija koje razmatramo
- Kažemo i da je simbol jednakosti bio **interpretiran**, dok su ostali simboli bili **neinterpretirani** ili **slobodni** simboli

Uvod i motivacija

Aksiome jednakosti – ponovo

- Drugi pogled na jednakosnu logiku je bio **sintaksno-deduktivni**: razmatrali smo formule koje su bile deduktivna posledica aksioma jednakosti
- Logičko opravdanje za ovakav pristup bila je činjenica da je formula F bila valjana u jednakosnoj logici akko je bilo moguće dokazati je polazeći od aksioma jednakosti
- Prednost sintaksno-deduktivnog pristupa je u tome što smo mogli da koristimo postojeće deduktivne sisteme i procedure u opštoj logici prvog reda, uz dodatak aksioma jednakosti
- Ipak, u praksi je bilo efikasnije koristiti specijalizovane procedure za rezonovanje sa jednakošću

Uvod i motivacija

Uopštenje?

- Uzmimo proizvoljan skup rečenica Ax nad izabranom signaturom \mathcal{L}
- Sintaksno-deduktivni pristup: posmatrajmo skup svih formula \mathcal{T} koje su deduktivna posledica skupa Ax
- Semantički pristup: posmatrajmo skup svih \mathcal{L} -struktura \mathcal{M} u kojima su tačne sve formule iz Ax
- Tada će i svaka formula iz \mathcal{T} biti tačna u svim strukturama iz \mathcal{M} (zbog saglasnosti deduktivnog sistema)
- Obrnuto: svaka formula koja je tačna u svim strukturama iz \mathcal{M} biće dokaziva iz Ax (zbog potpunosti deduktivnog sistema)
- Skup \mathcal{T} nazivaćemo **teorijom prvog reda** zadatom skupom **aksioma** Ax
- Strukture iz \mathcal{M} nazivaćemo **modelima** teorije \mathcal{T}

Teorija prvog reda

Definicija teorije prvog reda

- Teorija \mathcal{T} nad signaturom \mathcal{L} zadata skupom aksioma $Ax(\mathcal{T})$ je skup svih formula F takvih da je $Ax(\mathcal{T}) \vdash F$
- Formule $F \in \mathcal{T}$ zovemo **teoremama** teorije \mathcal{T} (u oznaci $\vdash_{\mathcal{T}} F$)
- **Deduktivna posledica** u teoriji (u oznaci $\Delta \vdash_{\mathcal{T}} F$): $Ax(\mathcal{T}) \cup \Delta \vdash F$
- **Model teorije**: struktura \mathcal{L} u kojoj su sve formule iz $Ax(\mathcal{T})$ tačne
- Formula F je **valjana u teoriji** \mathcal{T} (\mathcal{T} -valjana, u oznaci $\models_{\mathcal{T}} F$) ako je tačna u svim njenim modelima, tj. ako je $Ax(\mathcal{T}) \models F$
- Formula je **zadovoljiva u teoriji** \mathcal{T} (\mathcal{T} -zadovoljiva) ako je tačna u bar jednom modelu teorije \mathcal{T} , tj. ako je $Ax(\mathcal{T}) \cup \{F\}$ zadovoljiv skup formula
- **Logička posledica u teoriji** (u oznaci $\Delta \models_{\mathcal{T}} F$): F je tačna u svim modelima teorije \mathcal{T} u kojima su tačne sve formule iz Δ , tj. važi $Ax(\mathcal{T}) \cup \Delta \models F$

Teorija prvog reda

Veza između deduktivnih i semantičkih pojmova

Pod pretpostavkom da je deduktivni sistem koji razmatramo potpun i saglasan, važi:

- $\vdash_{\mathcal{T}} F$ akko $Ax(\mathcal{T}) \vdash F$ akko $Ax(\mathcal{T}) \models F$ akko $\models_{\mathcal{T}} F$
- $\Delta \vdash_{\mathcal{T}} F$ akko $Ax(\mathcal{T}) \cup \Delta \vdash F$ akko $Ax(\mathcal{T}) \cup \Delta \models F$ akko $\Delta \models_{\mathcal{T}} F$
- Rečenica F je \mathcal{T} -zadovoljiva akko nije $\models_{\mathcal{T}} \neg F$ tj. akko nije $\vdash_{\mathcal{T}} \neg F$
- $\vdash_{\mathcal{T}} F$ akko je $\neg F$ \mathcal{T} -nezadovoljiva

Teorija prvog reda

Osobine teorije prvog reda

- Za teoriju kažemo da je **aksiomatska** ako je zadata skupom aksioma $Ax(\mathcal{T})$ koji je rekurzivan (odlučiv)
- Za teoriju kažemo da je **potpuna** ako za svaku rečenicu A važi ili $\vdash_{\mathcal{T}} A$ ili $\vdash_{\mathcal{T}} \neg A$
- Za teoriju kažemo da je **konzistentna** ako ni za jednu rečenicu A ne važi istovremeno i $\vdash_{\mathcal{T}} A$ i $\vdash_{\mathcal{T}} \neg A$
- Za teoriju kažemo da je **odlučiva** ako postoji efektivan postupak koji za svaku rečenicu A u konačnom broju koraka ispituje da li je $\vdash_{\mathcal{T}} A$ ili ne

Teorija prvog reda

Mogu se dokazati sledeće važne činjenice

- Teorija je konzistentna akko ima model
- Za svaku konzistentnu teoriju koja ima bar jedan beskonačan model postoje modeli koji su neizomorfni (posledica Skolem-Lovenhajmove teoreme)
- Drugim rečima, u logici prvog reda se ne može postići **kategoričnost**, tj. ne može se formulirati konzistentna teorija koja ima tačno jedan model (do na izomorfizam). Za to su nam potrebne moćnije logike (poput logike drugog reda)
- Teorija je potpuna akko su joj svi modeli međusobno **elementarno ekvivalentni** (tj. za svaku rečenicu F važi da je F tačna u jednom modelu akko je tačna u drugom i obratno)
- Potpuna teorija je odlučiva akko je aksiomska
- Problem ispitivanja \mathcal{T} -zadovoljivosti formule F je odlučiv akko je teorija odlučiva u smislu prethodne definicije

Teorija prvog reda

Teorija kao skup modela

- Teorije se obično zadaju aksiomama koje nas ograničavaju na određeni skup modela koje razmatramo, a u kojima su sve aksiome (pa samim tim i teoreme) te teorije tačne
- Obrnuto, teorija se ponekad opisuje i zadavanjem skupa modela unapred
- Ključno pitanje je da li se za ovako zadatu teoriju može formulisati skup aksioma koji određuje upravo taj skup modela
 - na primer, u jednakosnoj logici smo podrazumevali skup svih normalnih modela
 - ipak, nismo mogli aksiomatski da opišemo baš taj skup modela
- Često se fiksira samo jedan model (npr. fiksiramo strukturu realnih brojeva ili strukturu prirodnih brojeva)
 - u takvim situacijama je jasno da ne postoji skup aksioma koji nas može ograničiti samo na taj jedan model, s obzirom na Skolem-Lovenhajmovu teoremu
- Ipak, ako zadata struktura zadovoljava sistem aksioma koji opisuje **potpunu** teoriju, tada je iz tog skupa aksioma moguće dokazati bilo koje tvđenje koje je tačno u zadatoj strukturi
- U suprotnom, ako ne postoji potpuna aksiomska teorija čiji je model data struktura, tada će za svaki konzistentan skup aksioma kome je model data struktura postojati tvđenja koja su tačna u toj strukturi, a koje se ne mogu dokazati iz tih aksioma
 - ovo je slučaj sa strukturom prirodnih brojeva (na osnovu **Gedelove teoreme o nepotpunosti aritmetike**)

Teorija prvog reda

Teorija prvog reda i jednakost

- Gotovo sve teorije koje su od značaja za razmatranje u matematici podrazumevaju postojanje relacije jednakosti
- Otuda ćemo mi u nastavku podrazumevati da simbol jednakosti uvek postoji, kao i da se interpretira baš kao relacija jednakosti
- Drugim rečima, podrazumevamo da su svi modeli teorije normalni modeli
- Alternativno, možemo podrazumevati da aksiome svake teorije uvek sadrže i aksiome jednakosti, čak i kada to nije eksplicitno navedeno

Primeri teorija prvog reda

Primeri

- Teorija skupova (ZFC) [neodlučiva]
- Peanova aritmetika [neodlučiva] (model \mathbb{N})
- Teorija grupa [neodlučiva]
- Teorija komutativnih grupa [odlučiva]
- Teorija polja [neodlučiva]
- Teorija uredjenih polja [neodlučiva] (modeli \mathbb{Q} , \mathbb{R})
- Teorija realno zatvorenih polja [odlučiva] (model \mathbb{R})
- Teorija algebarski zatvorenih polja [odlučiva] (model \mathbb{C})
- Euklidska geometrija [odlučiva] (model \mathbb{R}^3)

Rezonovanje u teoriji

Pristupi

- Opšte metode rezonovanja (poput rezolucije, uz dodate aksiome teorije)
 - Ovaj pristup je često neefikasan
- Specifične metode rezonovanja (posebne procedure koje uzimaju u obzir semantiku interpretiranih simbola teorije)
 - U praksi obično mnogo efikasnije
- Najčešće se dokazivanje teorema vrši pobijanjem, tj. dokazivanjem nezadovoljivosti negiranih formula

Razvoj specifičnih procedura za rezonovanje

Problemi

- Prilikom rezonovanja u teoriji, imamo tri problema koje treba da razmotrimo:
 - Tretman iskazne strukture formule
 - Tretman interpretiranih simbola u teoriji
 - Tretman kvantifikatora
- Raniji pristup za tretman iskazne strukture formule je bio svodjenje na DNF, pa razmatranje svake konjunkcije literala zasebno
- Moderni pristup je da se uposle efikasni SAT rešavači u tu svrhu
- Za tretman funkcijskih i predikatskih simbola teorije implementiraju se posebne procedure (uglavnom za bazni slučaj, tj. za konjunkcije baznih literala)

Rezonovanje u teoriji

Kvantifikatori

- Egzistencijalni kvantifikatori: **skolemizacija**
- Univerzalne kvantifikatore nije uvek moguće ukloniti
- **Rezolucija**: teško se kombinuje sa specifičnim procedurama koje uglavnom rade sa baznim formulama
- Neke teorije dopuštaju **eliminaciju kvantifikatora**
- Najčešći pristup je instanciranje kvantifikatora (zasnovan na Erbranovoj teoremi)
- Ovim se dobijaju bazne formule na koje se primenjuju specifične procedure odlučivanja

Furije-Mockinova procedura

Furije-Mockinova procedura

- Procedura odlučivanja za linearni fragment teorije uredjenih polja
- Modeli ove teorije su $(\mathbb{Q}, +, \leq)$ i $(\mathbb{R}, +, \leq)$ (linearne jednakosti i nejednakosti nad \mathbb{Q} ili \mathbb{R})
- Zasnovana na **eliminaciji kvantifikatora**

Furije-Mockinova procedura

Furije-Mockinova procedura

- Pretpostavimo da imamo konjunkciju K linearnih ograničenja oblika $a_1x_1 + a_2x_2 + \dots + a_nx_n \bowtie b$, gde je $\bowtie \in \{<, >, =\}$.
- Posmatrajmo, proizvoljnu fiksiranu promenljivu x_i . Ukoliko postoji bar jedna jednakost u kojoj se pojavljuje x_i , izrazimo je iz te jednakosti u obliku $x_i = \sum_{j \neq i} c_j x_j + c$
- Sada možemo u svim ostalim ograničenjima x_i zameniti izrazom $\sum_{j \neq i} c_j x_j + c$, a jednakost iz koje je izraženo x_i brišemo.
- Transformisana konjunkcija K' ne sadrži više promenljivu x_i
- Pritom, proizvoljna valuacija promenljivih $\{x_j \mid j \neq i\}$ će zadovoljavati K' akko se može proširiti nekom vrednošću za promenljivu x_i tako da dobijena valuacija zadovoljava originalnu konjunkciju K

Furije-Mockinova procedura

Furije-Mockinova procedura

- Ukoliko se promenljiva x_i ne pojavljuje ni u jednoj jednakosti (već samo u nejednakostima), tada sve nejednakosti koje sadrže x_i transformisati u jedan od oblika $x_i < \sum_{j \neq i} c_j x_j + c$ ili $x_i > \sum_{j \neq i} d_j x_j + d$
- Ako su sve nejednakosti prvog oblika (ili su sve drugog oblika) možemo ih sve obrisati
- U suprotnom, za svaku nejednakost prvog oblika i svaku nejednakost drugog oblika formiramo novu nejednakost $\sum_{j \neq i} d_j x_j + d < \sum_{j \neq i} c_j x_j + c$. Ove nejednakosti dodajemo u konjunktiju, a originalne nejednakosti koje sadrže x_i brišemo
- Slično kao i malopre, dobijena konjunktije K' će sadržati samo izraze koji ne sadrže x_i
- Takodje, K' će važiti u onim i samo onim valuacijama koje se mogu proširiti nekom vrednošću za x_i tako da bude zadovoljena originalna konjunktija K

Furije-Mockinova procedura

Furije-Mockinova procedura

- Pretpostavimo sada da imamo rečenicu u preneks normalnoj formi:
 $Q_1x_1.Q_2x_2.\dots.Q_nx_n.F$
- Možemo pretpostaviti da je Q_n egzistencijalni kvantifikator
- (Ukoliko Q_n nije egzistencijalni kvantifikator, formulu transformišemo u oblik $Q_1x_1.Q_2x_2.\dots.\neg\exists x_n.\neg F$)
- Transformišemo formulu F u DNF (prethodno sve nejednakosti oblika $l \leq r$ transformišemo u $(l < r \vee l = r)$, a sve različitosti $l \neq r$ transformišemo u $(l < r \vee r < l)$)
- Na svaku konjunciju u DNF-u primenimo prethodni postupak da eliminišemo promenljivu x_n
- Nakon toga se kvantifikator $\exists x_n$ može obrisati
- Postupak nastavljamo sa kvantifikatorom $Q_{n-1}x_{n-1}$ na isti način
- Na kraju dolazimo do formule koja ne sadrži promenljive ni kvantifikatore (bazna formula) koja je tačna akko je polazna formula bila teorema u teoriji uredjenih polja

Furije-Mockinova procedura

Furije-Mockinova procedura

- Opisana procedura se može koristiti i za ispitivanje zadovoljivosti konjunkcije baznih linearnih ograničenja
- Imajući u vidu skolemizaciju, sve (neinterpretirane) simbole konstanti koji se pojavljuju u konjunkciji možemo smatrati egzistencijalno kvantifikovanim promenljivama
- Otuda sve konstante možemo eliminisati jednu po jednu na ranije opisan način

Simpleks metod

Simpleks metod

- Simpleks metod je originalno osmišljen kao procedura za rešavanje problema **linearnog programiranja**
- U modernim alatima za automatsko rezonovanje se često koristi kao procedura za ispitivanje zadovoljivosti konjunkcije linearnih ograničenja nad poljem racionalnih brojeva

Simpleks metod

Simpleks metod

- Pretpostavimo da imamo konjunkciju linearnih ograničenja $a_{i1}x_1 + a_{i2}x_2 + \dots + a_{in}x_n \bowtie b_i$, gde je $\bowtie \in \{\leq, \geq\}$, $i \in 1 \dots m$
- Uvodimo nove konstante s_1, s_2, \dots, s_m , a originalni skup ograničenja transformišemo u sledeća dva skupa:
 - $s_j = a_{j1}x_1 + \dots + a_{jn}x_n$ (tablo)
 - $s_j \bowtie b_j$ (granična ograničenja)
- Tablo čine jednakosti kojima se neke nepoznate (inicijalno s_1, \dots, s_m) izražavaju preko ostalih nepoznatih (inicijalno x_1, \dots, x_n)
 - nepoznate koje se izražavaju u jednakostima se nazivaju **bazne nepoznate**
 - nepoznate preko kojih izražavamo bazne nepoznate se nazivaju **ne-bazne nepoznate**
- Tokom rada algoritma, skupovi baznih i ne-baznih nepoznatih se mogu menjati (ali uvek ostaju disjunktni).

Simpleks metod

Simpleks metod - inicijalizacija

- Svakoј nepoznatoј y se pridružuje vrednost $\beta(y)$ iz skupa racionalnih brojeva
- Takođe, svakoј nepoznatoј y se pridružuju njena donja i gornja granica $l(y)$ i $u(y)$
- Inicijalno, donja granica svih nepoznatih je $-\infty$, a gornja $+\infty$
- Na osnovu graničnih ograničenja se ove granice ažuriraju (npr. ako imamo ograničenje $y \leq 5$, tada $u(y)$ postavljamo na 5)
- Nakon toga vrednost svih ne-baznih nepoznatih postavljamo na vrednost neke od njihovih konačnih granica, ili na 0 ako su obe granice beskonačne
- Vrednost $\beta(y)$ bazne nepoznate y se izračunava na osnovu vrednosti ne-baznih nepoznatih i odgovarajuće relacije u tablu

Primedba

Primetimo da nakon ove inicijalizacije vrednosti svih ne-baznih nepoznatih zadovoljavaju svoja granična ograničenja, dok za bazne nepoznate to ne mora da važi.

Simpleks metod

Simpleks metod - algoritam

Pretpostavimo da je unapred zadat potpuni poredak \prec među nepoznatima. Označimo sa $a(x, y)$ koeficijent uz ne-baznu nepoznatu x u trenutnom stanju tabloa u jednačini kojom se izražava bazna nepoznata y . Algoritam se izvršava u sledećoj petlji:

- ako za sve bazne nepoznate važi $l(y) \leq \beta(y) \leq u(y)$, tada vratiti rezultat: **zadovoljivo**
- u suprotnom, neka je y najmanja bazna nepoznata (u poretku \prec) takva da ne zadovoljava svoja granična ograničenja (npr. neka je $\beta(y) > u(y)$)
- neka je x najmanja ne-bazna nepoznata (u poretku \prec) takva da važi $\beta(x) > l(x)$ (ako je $a(x, y) > 0$), odnosno $\beta(x) < u(x)$ (ako je $a(x, y) < 0$)
- ako ne postoji takvo x , tada vratiti rezultat: **nezadovoljivo**
- u suprotnom, iz jednačine kojom se izražava y u tabloju izraziti x , a zatim u svim drugim jednačinama zameniti x dobijenom desnom stranom
- na ovaj način, y postaje ne-bazna nepoznata, a x postaje bazna nepoznata (ovo je poznato i kao **pivotiranje**)
- nakon toga, ne-baznoj nepoznatoj y postaviti vrednost na $u(y)$
- zatim ažurirati vrednosti svih baznih nepoznatih na osnovu ove promene
- vratiti se na početak petlje i ponoviti postupak

Simpleks metod

Ograničenja $<$, $>$, $=$, \neq

- Simpleks algoritam radi samo sa relacijama \leq , \geq
- Jednakost $u = c$ se može zameniti sa $u \leq c$ i $u \geq c$
- Stroga nejednakost $u < c$ se može zameniti sa $u \leq c - \varepsilon$, gde je $\varepsilon > 0$ dovoljno mala pozitivna vrednost (slično za $u > c$)
- Različitost $u \neq c$ se može zameniti disjunkcijom $u < c \vee u > c$, tj. razmatraju se dva podproblema

Simpleks metod

Celobrojna ograničenja

- Pretpostavimo da za neke nepoznate dodatno zahtevamo da moraju imati celobrojne vrednosti
- Najpre se razmatra **racionalna relaksacija** ovog problema, tj. uslovi celobrojnosti se ignorišu i traži se bilo kakvo racionalno rešenje
- Ako je relaksacija problema nezadovoljiva, tada je nezadovoljiv i polazni problem
- U suprotnom, za svaku nepoznatu y za koju se zahteva da bude celobrojna, proveravamo da li je dobijena vrednost $\beta(y)$ zaista ceo broj
- ako je taj uslov ispunjen za svaku takvu promenljivu, tada je polazni problem zadovoljiv
- u suprotnom, razmatramo dva podproblema:
 - prvi uz dodatno ograničenje $y \leq \lfloor \beta(y) \rfloor$
 - drugi uz dodatno ograničenje $y \geq \lceil \beta(y) \rceil$
- Ako je bilo koji od ova dva problema zadovoljiv, zadovoljiv je i polazni problem
- Ova tehnika poznata je i kao tehnika **grananja sa ograničavanjem** (engl. **branch-and-bound**)
- Tehnika se može kombinovati sa raznim tehnikama zasnovanim na ravnima odsecanja (engl. **cutting plane**), radi brže konvergencije

Simpleks metod

Simpleks metod

- Složenost simpleks metoda je u najgorem slučaju eksponencijalna
- Ipak, u praksi često daje dobre rezultate
- Primenjiva i u racionalnom i u celobrojnom slučaju (uz dodatne tehnike)
- Poredak \prec je bitan da bi se obezbedilo zaustavljanje
- Dobar izbor poretka može u značajnoj meri da ubrza konvergenciju
- Dobra strana algoritma je **inkrementalnost**: ako dodamo još neka granična ograničenja naknadno, možemo prosto ažurirati granice i ponovo pokrenuti algoritam

Pregled

- 1 Teorije prvog reda
- 2 SMT problem i SMT rešavači

SMT problem i SMT rešavači

SMT problem

- SMT problem (engl. Satisfiability Modulo Theory) za teoriju \mathcal{T} je problem ispitivanja \mathcal{T} -zadovoljivosti date formule F
- Odlučivost SMT problema zavisi od izbora teorije \mathcal{T}
- Ispitivanje valjanosti formule F u teoriji se svodi na SMT problem za formulu $\neg F$ u toj teoriji
- Za pojedine teorije, SMT problem je odlučiv samo za neke fragmente (formule određenog oblika)

SMT rešavači

- Softverski alati koji implementiraju procedure odlučivanja za odlučive SMT probleme zovu se SMT rešavači
- Relativno nova tehnologija (početak 21. veka)
- SAT tehnologija + instanciranje kvantifikatora + bazne procedure odlučivanja (lenji pristup)
- Primene: verifikacija softvera i hardvera, problemi planiranja, problemi zadovoljavanja ograničenja...

Višesortna logika prvog reda

Motivacija

- Uobičajena definicija logike prvog reda podrazumeva jedinstven domen (univerzum) iz koga se uzimaju vrednosti kojima se interpretiraju svi termovi
- Dakle, svi izrazi su ili termovi ili formule, pri čemu se svi termovi interpretiraju elementima iz tog jedinstvenog domena, dok se formule interpretiraju kao **tačne** ili **netačne**
- Ovakva definicija onemogućava razlikovanje tipova
- U mnogim primenama, poželjno je razlikovati tipove (npr. ako semantiku nekog programa opisujemo u okviru logike prvog reda, želimo da razlikujemo npr. izraze tipa `int` od izraza tipa `double`).
- Dakle, poželjno je termove razlikovati po **sortama**, pri čemu za svaku sortu termova postoji poseban domen iz koga se uzimaju vrednosti
- Ovakav logički okvir se naziva **višesortna logika prvog reda**
- Naročito se razmatra u kontekstu SMT problema

Sintaksa

Signatura $\mathcal{L} = (\mathcal{S}, \mathcal{F}, r)$

- Skup sorti \mathcal{S} (među kojima je i Bool)
- Skup funkcijskih simbola \mathcal{F} (među kojima su i \top , \perp , \wedge , \vee , \neg ...)
- Funkcija ranga $r : \mathcal{F} \rightarrow \mathcal{S}^* \times \mathcal{S}$
- $r(f) = [s_1, \dots, s_n] \rightarrow s$ (s_1, \dots, s_n su sorte argumenata, a s je povratna sorta)
- $r(a) = [] \rightarrow s$ (a je simbol konstante sorte s)
- Za svaku sortu $s \in \mathcal{S}$, skup promenljivih V^s

Izrazi

- Svaki izraz e ima svoju sortu s
- Simbol konstante a ranga $[] \rightarrow s$ je izraz sorte s
- Promenljiva $x \in V^s$ je izraz sorte s
- Ako je t_i izraz sorte s_i ($i = 1, \dots, n$) i $r(f) = [s_1, \dots, s_n] \rightarrow s$, tada je $f(t_1, \dots, t_n)$ izraz sorte s
- Izraze sorte Bool zovemo **formulama**, a ostale izraze **termovima**
- Iskazni simboli imaju uobičajene rangove: $r(\perp) = r(\top) = [] \rightarrow \text{Bool}$, $r(\neg) = [\text{Bool}] \rightarrow \text{Bool}$, $r(\wedge) = [\text{Bool}, \text{Bool}] \rightarrow \text{Bool}$, ...
- Simbol jednakosti: $r(=) = [s, s] \rightarrow \text{Bool}$ za svaku sortu $s \in \mathcal{S}$
- Promenljive se u formulama mogu kvantifikovati: $\forall x : \sigma.F$ ili $\exists x : \sigma.F$, gde je x promenljiva sorte σ , a F je formula

Semantika

\mathcal{L} -struktura $\mathcal{M} = (\mathcal{D}, \cdot^{\mathcal{M}})$

- $\mathcal{D} = \{D^s \mid s \in \mathcal{S}\}$: skup domena (za svaku sortu s po jedan)
- $D^{\text{Bool}} = \{0, 1\}$: domen sorte Bool je uvek $\{0, 1\}$
- $a^{\mathcal{M}} \in D^s$ ($r(a) = [] \rightarrow s$): konstanta sorte s uzima vrednost iz D^s
- $f^{\mathcal{M}} : D^{s_1} \times \dots \times D^{s_n} \rightarrow D^s$ ($r(f) = [s_1, \dots, s_n] \rightarrow s$)
- Uobičajeno značenje iskaznih simbola: $\perp^{\mathcal{M}} = 0$, $\top^{\mathcal{M}} = 1$, $\neg^{\mathcal{M}}(x) = 1$ akko je $x = 0$, $\wedge^{\mathcal{M}}(x, y) = 1$ akko $x = 1$ i $y = 1$, ...
- $=_s^{\mathcal{M}}(x, y) = 1$ akko su x i y isti element skupa D^s

Interpretacija

- Valuacija $v : V^s \rightarrow D^s$ (za svako $s \in \mathcal{S}$)
- $x \in V^s$: $I_v^{\mathcal{M}}(x) = v(x)$
- $r(a) = [] \rightarrow s$: $I_v^{\mathcal{M}}(a) = a^{\mathcal{M}}$
- $r(f) = [s_1, \dots, s_n] \rightarrow s$: $I_v^{\mathcal{L}}(f(t_1, \dots, t_n)) = f^{\mathcal{M}}(I_v^{\mathcal{L}}(t_1), \dots, I_v^{\mathcal{L}}(t_n))$
- $I_v^{\mathcal{M}}(\forall x : \sigma.F) = 1$ akko $I_{v'}^{\mathcal{M}}(F) = 1$ za svako v' ($v'(y) = v(y)$ za $y \neq x$)
- $I_v^{\mathcal{M}}(\exists x : \sigma.F) = 1$ akko $I_{v'}^{\mathcal{M}}(F) = 1$ za neko v' ($v'(y) = v(y)$ za $y \neq x$)
- Interpretacija zatvorenih formula (rečenica) ne zavisi od v ($I_v^{\mathcal{M}}(F) = I^{\mathcal{M}}(F)$)

Semantika – nastavak

Osnovne semantičke pojmove definišemo na uobičajen način

- $\mathcal{M} \models F$: zatvorena formula F je tačna u \mathcal{L} -strukturi \mathcal{M} ($I^{\mathcal{M}}(F) = 1$)
- Zadovoljiva formula: postoji \mathcal{L} -struktura \mathcal{M} takva da je $\mathcal{M} \models F$
- $\models F$: valjana formula (tačna u svim \mathcal{L} -strukturama)
- $\Delta \models F$: logička posledica (tačna kad god su tačne sve formule iz Δ)
- $F_1 \equiv F_2$: logička ekvivalencija ($F_1 \models F_2$ i $F_2 \models F_1$)
- $F \models \perp$: nezadovoljiva (negacija $\neg F$ je valjana formula)

Napomene

Napomene

- Višesortna logika tretira sve simbole na uniforman način
- Sorta Bool je samo jedna od sorti (fiksiranog značenja)
- Predikatski simboli su prosto funkcijski simboli koji vraćaju Bool
- Iskazni simboli \top , \perp , \neg , \wedge , \vee , ... su samo funkcijski simboli nad Bool sortom i sa fiksnim značenjem

Izražajnost

- Višesortna logika prvog reda **nema** veću izražajnost od uobičajene logike prvog reda bez sorti
- Za svaku \mathcal{L} -strukturu \mathcal{M} (u smislu višesortne logike) smo u standardnoj logici prvog reda mogli uzeti strukturu čiji je univerzum **unija** domena svih sorti iz \mathcal{L}
- Postojanje više sorti se sada može simulirati dodatnim unarnim predikatskim simbolima p^s koji se interpretiraju kao odgovarajući podskupovi univerzuma koji odgovaraju domenima pojedinih sorti $s \in \mathcal{L}$

EUF teorija

EUF teorija

- Equality with Uninterpreted Functions
- Nema drugih predikatskih simbola osim jednakosti (svi atomi su oblika $u = v$)
- Signatura može saržati proizvoljan broj sorti i funkcijskih simbola koji se mogu potpuno slobodno interpretirati (neinterpretirani simboli i sorte)
- Teorija nema aksioma (osim aksioma jednakosti)
- Modeli teorije su sve (normalne) interpretacije
- SMT problem za ovu teoriju je neodlučiv
- SMT problem za bazni fragment ove teorije (u oznaci QF_UF) je odlučiv
- Zadovoljivost konjunkcije baznih literala ove teorije je odlučiv u polinomskom vremenu (procedure zasnovane na kongruentnim zatvorenjima, poput Nelson-Openove procedure)

Realna aritmetika

Realna aritmetika

- Signatura: sorta `Real`, simboli $0, 1, +, \cdot, -, /, \leq$
- Aksiome: realna zatvorena polja
- Standardni model: struktura realnih brojeva \mathbb{R}
- Teorija je odlučiva
- Procedura odlučivanja: Cilindrična algebarska dekompozicija (CAD) (eliminacija kvantifikatora)
- Bazni linearni fragment (u oznaci `QF_LRA`)
- Problem ispitivanja zadovoljivosti konjunkcije linearnih baznih literala je odlučiv u polinomskom vremenu
- Neke od procedura odlučivanja (eksponencijalne složenosti) su Furije-Mockinova procedura (zasnovana na eliminaciji kvantifikatora) i Simpleks procedura

Celobrojna aritmetika

Celobrojna aritmetika

- Signatura: sorta Int , simboli $0, 1, +, \cdot, -, \leq$
- Standardni model: struktura celih brojeva \mathbb{Z}
- Teorija je neodlučiva
- Njen linearni fragment je odlučiv
- Bazni linearni fragment (QF_LIA)
- Problem ispitivanja zadovoljivosti konjunkcije linearnih baznih literala je odlučiv i NP-kompletan
- Procedure odlučivanja: simpleks + tehnike odsecanja

Teorija nizova

Teorija nizova

- Signatura: sorte Index, Value i Array, simboli
 - $select : [Array, Index] \rightarrow Value$
 - $store : [Array, Index, Value] \rightarrow Array$
- Aksiome:
 - $\forall x. \forall y. \forall z. select(store(x, y, z), y) = z$
 - $\forall x. \forall y_1. \forall y_2. \forall z. y_1 \neq y_2 \Rightarrow select(store(x, y_1, z), y_2) = select(x, y_2)$
 - $\forall x_1. \forall x_2. (\forall y. select(x_1, y) = select(x_2, y)) \Rightarrow x_1 = x_2$
- Teorija je neodlučiva u opštem slučaju
- Bazni fragment teorije (QF_AX) je odlučiv
- Problem ispitivanja zadovoljivosti konjunkcije baznih literala je NP-kompletan
- Procedure odlučivanja: kongruento zatvorenje + instanciranje aksioma

Teorija bitvektora

Teorija bitvektora

- Signatura: sorte BitVec_n ($n \in \mathbb{N}$), simboli:
 - $\text{bvnot}_n, \text{bvneg}_n : [\text{BitVec}_n] \rightarrow \text{BitVec}_n$
 - $\text{bvadd}_n, \text{bvshl}_n, \dots : [\text{BitVec}_n, \text{BitVec}_n] \rightarrow \text{BitVec}_n$
 - $\text{bvult}_n, \text{bvslt}_n, \dots : [\text{BitVec}_n, \text{BitVec}_n] \rightarrow \text{Bool}$
- Standardni model: hardverska aritmetika
- Teorija je odlučiva
- Bazni fragment QF_BV
- Problem ispitivanja zadovoljivosti konjunkcije baznih literala je NP-kompletan
- Procedure odlučivanja: svodjenje na SAT rastavljanjem na bitove (bit-blasting)

Teorija rekurzivnih (induktivnih) tipova podataka

Signatura

- skupa sorti koje predstavljaju rekurzivne tipove podataka
 - npr. *Nat* i *List*
- skupa konstruktora podataka za ove tipove, npr.:
 - $zero : [] \rightarrow Nat$, $succ : [Nat] \rightarrow Nat$,
 - $empty : [] \rightarrow List$, $cons : [Nat, List] \rightarrow List$
- skupa selektora podataka za ove tipove, npr.:
 - $pred : [Nat] \rightarrow Nat$
 - $head : [List] \rightarrow Nat$, $tail : [List] \rightarrow List$
- skupa testera za ove tipove, npr.:
 - $is_zero : [Nat] \rightarrow Bool$, $is_succ : [Nat] \rightarrow Bool$
 - $is_empty : [List] \rightarrow Bool$, $is_cons : [List] \rightarrow Bool$

Teorija rekurzivnih (induktivnih) tipova podataka (2)

Aksiome

- različito konstruisani objekti su različiti, npr.:
 - $\forall x: \text{Nat}. \text{zero} \neq \text{succ}(x), \forall x: \text{Nat } y: \text{Nat}. x \neq y \Rightarrow \text{succ}(x) \neq \text{succ}(y)$
 - $\forall l: \text{List } x: \text{Nat}. \text{empty} \neq \text{cons}(x, l),$
 $\forall l_1: \text{List } x_1: \text{Nat } l_2: \text{List } x_2: \text{Nat}. l_1 \neq l_2 \vee x_1 \neq x_2 \Rightarrow \text{cons}(x_1, l_1) \neq \text{cons}(x_2, l_2)$
- aksiome za selektore, npr.:
 - $\forall x: \text{Nat}. \text{pred}(\text{succ}(x)) = x$
 - $\forall x: \text{Nat } l: \text{List}. \text{head}(\text{cons}(x, l)) = x, \forall x: \text{Nat } l: \text{List}. \text{tail}(\text{cons}(x, l)) = l$
- aksiome za testere, npr.:
 - $\text{is_zero}(\text{zero}) = \text{True}, \forall x: \text{Nat}. \text{is_zero}(\text{succ}(x)) = \text{False}, \text{is_succ}(\text{zero}) = \text{False},$
 $\forall x: \text{Nat}. \text{is_succ}(\text{succ}(x)) = \text{True}$
 - $\text{is_empty}(\text{empty}) = \text{True}, \forall x: \text{Nat } l: \text{List}. \text{is_empty}(\text{cons}(x, l)) = \text{False},$
 $\text{is_cons}(\text{empty}) = \text{False}, \forall x: \text{Nat } l: \text{List}. \text{is_cons}(\text{cons}(x, l)) = \text{True}$
- aksiomatske sheme indukcije (za svaku formulu ϕ), npr.:
 - $(\phi(\text{zero}) \wedge (\forall x: \text{Nat}. \phi(x) \Rightarrow \phi(\text{succ}(x)))) \Rightarrow (\forall x: \text{Nat}. \phi(x))$
 - $(\phi(\text{empty}) \wedge (\forall x: \text{Nat } l: \text{List}. \phi(l) \Rightarrow \phi(\text{cons}(x, l)))) \Rightarrow (\forall l: \text{List}. \phi(l))$

Teorija rekurzivnih (induktivnih) tipova podataka (3)

Odlučivost i procedure odlučivanja

- teorija je neodlučiva
- bazni fragment je **odlučiv**
- procedura odlučivanja: kongruentno zatvorenje + sistem pravila koji oponaša aksiome teorije

Kombinacije teorija

Teorija $\mathcal{T}_1 + \mathcal{T}_2 + \dots + \mathcal{T}_n$

Neka su date teorije $\mathcal{T}_1, \mathcal{T}_2, \dots, \mathcal{T}_n$ nad signaturama $\Sigma_1, \Sigma_2, \dots, \Sigma_n$, respektivno. **Kombinacija teorija** $\mathcal{T} = \mathcal{T}_1 + \mathcal{T}_2 + \dots + \mathcal{T}_n$ je teorija nad signaturom $\Sigma_1 \cup \Sigma_2 \cup \dots \cup \Sigma_n$, čiji se skup modela \mathcal{M} sastoji iz svih struktura \mathcal{D} takvih da je projekcija \mathcal{D}_i te strukture na Σ_i model teorije \mathcal{T}_i .

Napomena

Ukoliko su teorije $\mathcal{T}_1, \dots, \mathcal{T}_n$ zadate skupovima aksioma $Ax(\mathcal{T}_1), \dots, Ax(\mathcal{T}_n)$, tada će njihova kombinacija \mathcal{T} biti upravo teorija zadata skupom aksioma $Ax(\mathcal{T}_1) \cup \dots \cup Ax(\mathcal{T}_n)$.

Kombinacije teorija (2)

Kombinacije teorija u SMT-u

U SMT-u se razmatraju različite kombinacije ranije opisanih teorija:

- QF_UFLRA: kombinacija linearne realne aritmetike i EUF teorije
- QF_UFLIA: kombinacija linearne celobrojne aritmetike i EUF teorije
- QF_AUFLIA: kombinacija teorije nizova, linearne celobrojne aritmetike i EUF teorije
- QF_AUFBV: kombinacija teorije nizova, teorije bitvektora i EUF teorije
- ...

Kombinacije teorija (3)

Procedure odlučivanja za kombinacije teorija

- Pod određenim uslovima, procedure odlučivanja za kombinacije teorija se mogu dobiti kombinovanjem procedura odlučivanja za teorije koje se kombinuju
- Obično se zahteva da važe sledeći uslovi:
 - da su signature teorija koje se kombinuju međusobno disjunktne (do na simbol jednakosti i određeni skup **deljenih konstanti**)
 - da su teorije koje se kombinuju **stabilno beskonačne**
- Teorija \mathcal{T} je **stabilno beskonačna** ako je svaka \mathcal{T} -zadovoljiva rečenica tačna u nekom beskonačnom modelu teorije \mathcal{T}
- Umesto stabilne beskonačnosti se ponekad razmatraju i nešto slabiji uslovi
- Postoji više shema za kombinovanje procedura odlučivanja:
 - **Nelson-Openova** shema kombinovanja
 - **Odloženo kombinovanje teorija** (engl. **Delayed Theory Combination (DTC)**)

SMT rešavači

Lenji pristup

- Iskazna apstrakcija: (bazni) atomi prvog reda se zamenjuju iskaznim slovima
- SAT rešavač ispituje zadovoljivost dobijene iskazne formule
- Zadovoljavajuća iskazna valuacija određuje konjunkciju baznih literala
- Posebna procedura odlučivanja proverava zadovoljivost dobijene konjunkcije baznih literala u teoriji
- Rezultat: efikasnost pretrage SAT rešavača + procedure odlučivanja prilagođene teoriji
- „Lenja DNF transformacija”
- Kvantifikatori: skolemizacija (egzistencijalni) + instanciranje (univerzalni)
- Tretman kvantifikatora je najčešće nepotpun (primenjuju se razne heuristike)

CDCL(\mathcal{T})

CDCL(\mathcal{T}) (Nievenhuis, Oliveras, Tineli (2006))

- Najčešće korišćena arhitektura zasnovana na lenjom pristupu
- CDCL zasnovan SAT rešavač + \mathcal{T} -rešavač
- SAT rešavač: inkrementalno konstruiše zadovoljavajuće iskazne valuacije
- \mathcal{T} -rešavač: ispituje zadovoljivost odgovarajuće konjunkcije literala prvog reda u teoriji \mathcal{T} u toku konstrukcije zadovoljavajuće valuacije
- Mogućnost rezonovanja *unapred* u teoriji (teorijske propagacije)

CDCL(\mathcal{T})

CDCL(X) – sistem zasnovan na pravilima

- Implementira klasičan CDCL zasnovan SAT rešavač proširen dodatnim pravilima za rezonovanje u teoriji
- Stanje (F, M, C) : F skup klauza, M je stek literala (parcijalna valuacija), C je konfliktni skup (ili *no_cflt* ako nema konflikta)
- Pravilo: definiše način promene stanja, kao i uslove pod kojima se može primeniti
- Grananje, jedinična propagacija, analiza konflikata i nechronološko vraćanje unazad
- Dodatno: teorijske propagacije i konflikti

Funkcionalnost \mathcal{T} -rešavača

Obavezna funkcionalnost

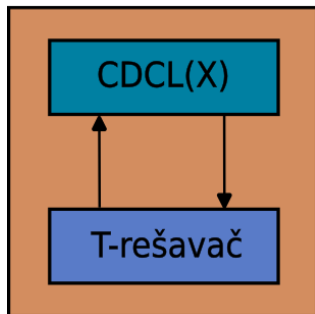
- da može da utvrdi da li je konjunkcija literala na steku zadovoljiva u teoriji
- da može da konstruiše objašnjenje konflikta ($R \subset M$ takav da je $R \models_{\mathcal{T}} \perp$)

Poželjna funkcionalnost

- da može da vrši teorijske propagacije i generiše njihova objašnjenja ($R \subset M$ takav da je $R \models_{\mathcal{T}} l$, gde je l propagirani literal)
- inkrementalnost (mogućnost efikasne provere zadovoljivosti u slučaju dodavanja novih literala u konjunkciju bez pokretanja celog postupka iz početka)
- efikasna rekonstrukcija prethodnog stanja (za potrebe vraćanja unazad)

CDCL(\mathcal{T})

CDCL(\mathcal{T}) zasnovan SMT rešavač



Struktura

- SMT rešavač ima modularnu strukturu, komponente su jasno odvojene i komuniciraju putem precizno definisanog interfejsa
- ovakva arhitektura omogućava da se \mathcal{T} -rešavač zameni \mathcal{T}' -rešavačem za neku drugu teoriju \mathcal{T}' bez ikakvih promena na SAT rešavaču
- $CDCL(X) + \mathcal{T}$ -rešavač = $CDCL(\mathcal{T})$

Primer interfejsa teorijskog rešavača

Interfejs procedure

- *newLevel()* – uspostavljanje novog nivoa odlučivanja
- *backtrack(m)* – vraćanje unazad na nivo m
- *assert(l)* – dodavanje literala l na stek
- *checkConflict(E)* – provera konflikta u teoriji
- *checkPropagate(L)* – detekcija teorijskih propagacija
- *explainLiteral(l, E)* – objašnjavanje propagiranog literala

SMT-LIB

SMT-LIB

- Poznati SMT rešavači: Z3, Yices, CVC, MathSAT, OpenSMT, BarcelogicTools
- Cilj SMT-LIB inicijative: bolja koordinacija u razvoju i lakše poređenje SMT rešavača
- Standard SMT-LIB (tekuća verzija 2.6): ulazno-izlazni jezik, logički okvir, teorije, kombinacije teorija
- Velika biblioteka instanci za testiranje i poređenje
- <http://smtlib.cs.uiowa.edu/>