

Shema dva posmatrana literala u DPLL SAT

rešavačima

Seminarski rad u okviru kursa
Automatsko rezonovanje
Matematički fakultet

Staša Đorđević 1007/2025

6. februar 2026.

Sažetak

TODO sazetak na kraju - rezime rada, max 100 reci

1 Uvod

Automatsko rezonovanje je oblast računarske nauke koja se bavi razvojem metoda i algoritama za automatsko izvođenje zaključaka iz formalnih logičkih sistema. Jedan od centralnih problema u ovoj oblasti je problem zadovoljivosti iskazne logike (SAT problem), koji se sastoji u određivanju da li postoji interpretacija kojom se formula iskazne logike može učiniti istinitom. SAT problem je fundamentalni NP-kompletan problem i predstavlja osnovu za mnoge primene u verifikaciji softvera, veštačkoj inteligenciji i optimizaciji.

Jedna od najefikasnijih metoda za rešavanje SAT problema je DPLL algoritam (Davis–Putnam–Logemann–Loveland), koji predstavlja unapređenje klasičnog DP (Davis–Putnam) postupka. DPLL algoritam kombinuje sistematsko pretraživanje sa jednostrukim dedukcijama na osnovu jedinica (unit propagation) i detekcijom konflikata, čime omogućava efikasno ispitivanje velikih formula. Ključni deo DPLL algoritma predstavlja izbor literalu i način na koji se donose odluke tokom pretraživanja, jer od toga zavisi brzina pronalaženja rešenja ili dokaza da rešenje ne postoji.

U okviru ovog rada fokus je stavljen na analizu i implementaciju *sheme dva posmatrana literala* unutar DPLL SAT rešavača. Ova shema predstavlja strategiju odabira literalu, koja istovremeno razmatra dva kandidata za sledeći izbor, čime se povećava verovatnoća bržeg pronalaženja rešenja i smanjuje broj potrebnih grananja pretraživanja. Razumevanje i pravilna implementacija ove sheme može značajno poboljšati performanse DPLL algoritma, posebno na velikim i složenim SAT instancama.

Ostatak ovog rada organizovan je na sledeći način. U poglavljju 2 uvodimo osnovne definicije i notaciju iz iskazne logike, kao i formalni opis SAT problema. Poglavlje 3 sadrži detaljan opis DPLL algoritma i sheme dva posmatrana literala. Poglavlje 4 opisuje implementaciju ove metode u izabranom programskom jeziku Python, uključujući ključne klase i funkcije. Na kraju, poglavljje 5 daje osvrt na rezultate rada i diskusiju o efikasnosti primenjene strategije.

2 Osnove

U ovom poglavlju uvodimo osnovne pojmove i notaciju iz iskazne logike koji će biti korišćeni u ostatku rada.

2.1 Sintaksa iskazne logike

Iskazna logika se bazira na *atomskim iskazima*, koji predstavljaju osnovne logičke promenljive, i na *logičkim veznicima* koji povezuju iskaze u složenije formule.

Formule iskazne logike definišu se rekurzivno:

- Iskazna slova i logičke konstante su iskazne formule.
- Ako su F i G formule, onda su i (F) , $\neg F$ (negacija), $F \wedge G$ (konjunkcija), $F \vee G$ (disjunkcija), $F \rightarrow G$ (implikacija) i $F \leftrightarrow G$ (ekvivalencija) formule.

2.2 Semantika iskazne logike

Semantika iskaznih formula definiše se pomoću pojma valuacije i interpretacije. Valuacija je funkcija $v : P \rightarrow \{0, 1\}$ koja svakom atomu iz skupa iskaznih slova P dodeljuje logičku vrednost.

Svaka (potpuna) valuacija v indukuje funkciju $I_v : F_P \rightarrow \{0, 1\}$ na skupu svih iskaznih formula nad P (u oznaci F_P), koju nazivamo interpretacija. Ova funkcija svakoj formuli pridružuje istinitosnu vrednost i definiše se rekurzivno na sledeći način:

- $I_v(p) = 1$ akko je $v(p) = 1$;
- $I_v(\top) = 1$, $I_v(\perp) = 0$;
- $I_v(\neg F) = 1$ akko je $I_v(F) = 0$;
- $I_v(F_1 \wedge F_2) = 1$ akko je $I_v(F_1) = 1$ i $I_v(F_2) = 1$;
- $I_v(F_1 \vee F_2) = 1$ akko je $I_v(F_1) = 1$ ili $I_v(F_2) = 1$;
- $I_v(F_1 \Rightarrow F_2) = 1$ akko je $I_v(F_1) = 0$ ili $I_v(F_2) = 1$;
- $I_v(F_1 \Leftrightarrow F_2) = 1$ akko je $I_v(F_1) = I_v(F_2)$.

2.3 Zadovoljivost i tautologija

Formula F je *zadovoljiva* ako postoji barem jedna interpretacija I takva da je $I(F) = \text{True}$. Formula je *tautologija* ako je $I(F) = \text{True}$ za sve moguće interpretacije I . S druge strane, formula je *nezadovoljiva* ako ne postoji nijedna interpretacija koja je čini istinitom.

2.4 SAT problem

Problem zadovoljivosti iskazne logike (SAT problem) je zadatak određivanja da li je data formula F zadovoljiva. SAT problem je NP-kompletan i predstavlja osnovni problem u oblasti automatskog rezonovanja.

U praksi, često se koriste formule u *konjunktivnoj normalnoj formi* (CNF), gde je formula predstavljena kao konjunkcija klauza, a svaka klauza je disjunkcija literala. Literal je ili atomski iskaz ili njegova negacija.

3 Opis metode

Ovo poglavlje treba da sadži detaljniji opis samog algoritma koji je implementiran u okviru seminar skog rada. Algoritam može biti opisan rečima ili pseudokodom (ne detaljno, već samo na nivou strukture). Treba poseban akcenat staviti na ključne delove algoritma i njihovu ulogu. Poželjno je navesti i primere koji dodatno razjašnjavaju rad algoritma.

4 Implementacija

U ovom poglavlju treba dati detalje implementacije. Treba navesti u kom programskom jeziku je metoda implementirana, kako je kôd organizovan, koje su ključne funkcije/klase/metode i koja je njihova uloga.

Takodje je bitno dati upustvo za prevodjenje i pokretanje projekta, kao i navesti softver koji je potreban za prevodjenje i pokretanje programa (prevodilac, alati, dodatne biblioteke i sl.).

5 Zaključak

U zaključku treba dati osvrt na ceo rad, kao i na metodu koja je u njemu prikazana. Ceo rad ne treba da ima više od 10 strana. Nakon zaključka, navesti spisak relevantne literature, ako je ista korišćena u izradi seminar skog. Svi navedeni radovi iz spiska moraju biti citirani negde u tekstu na ovaj način [2]. Primer spiska literature je dat u nastavku (primer knjige [2], rada na konferenciji [3] i rada u časopisu [4]).

DPLL (Davis-Putnam-Logemann-Loveland) procedura ispituje zadovoljivost iskaznih formula datih u KNF obliku. Ova procedura je osmisljena 1962. godine od strane istraživaca po kojima je i dobila ime. Ova procedura predstavlja proceduru pretrage, što znači da će vratiti valvaciju koja zadovoljava formulu u slučaju da je ona zadovoljiva. DPLL procedura je nastala kao nadogradnja DP procedure koja je zauzimala vise memorije. Samim tim, nasledila je dosta koraka iz DP procedure. DP procedura je sustinski zaznavana na rezoluciji i ima tri osnovne operacije:

- propagacija jedinicnih klauza (unit propagate)
- eliminacija cistih literala (pure literal)
- eliminacija promenljive (variable elimination)

Literatura

- [1] Prezentacija sa kursa Automatsko rezonovanje <https://poincare.matf.bg.ac.rs/~milan.bankovic/preuzimanje/ar/ar-iskazna-logika.pdf>
- [2] Biere, Armin, Marijn Heule, and Hans van Maaren, eds. Handbook of satisfiability. Vol. 185. IOS press, 2009.
- [3] Ganzinger, Harald, et al. *DPLL (T): Fast decision procedures*. Computer Aided Verification: 16th International Conference, CAV 2004, Boston, MA, USA, July 13-17, 2004. Proceedings 16. Springer Berlin Heidelberg, 2004.
- [4] Nieuwenhuis, Robert, Albert Oliveras, and Cesare Tinelli. *Solving SAT and SAT modulo theories: From an abstract Davis–Putnam–Logemann–Loveland procedure to DPLL (T)*. Journal of the ACM (JACM). 2006.