

Izražajnost + Automatizacija + Ispravnost: Kombinovanje SMT rešavača i interaktivnih dokazivača teorema

Seminarski rad u okviru kursa
Verifikacija softvera
Matematički fakultet

Miloš Samardžija
miloss208@gmail.com

14. maj 2018.

Sažetak

Ovaj tekst predstavlja sažetak naučnog rada "Expressiveness + Automation + Soundness: Towards Combining SMT Solvers and Interactive Proof Assistants"[1]. Prikazan je značaj SMT rešavača u ulozi pomoćnog alata prilikom interaktivnog dokazivanja teorema. U radu je opisana kombinacija interaktivnog dokazivača teorema - Isabelle, i SMT rešavača koji sadrži SAT rešavač i procedure odlučivanja za logiku prvog reda bez kvantifikatora sa jednakostima.

Sadržaj

1	Uvod	2
2	Motivacija za integraciju alata	3
3	Rekonstrukcija dokaza	3
4	Zaključak	5
	Literatura	5

1 Uvod

Formalni razvoj sistema zahteva izražajne jezike za pisanje specifikacije, ali i alate sa visokim stepenom automatizacije. Kombinovanje ova dva cilja nije jednostavno, posebno ako se pored toga zahteva i vrlo visoka garancija ispravnosti.

Deduktivni alati za verifikaciju sistema mogu biti klasifikovani prema njihovoj izražajnosti, stepenu automatizacije i garanciji ispravnosti. Idealan alat bi trebalo da se pokaže dobro u sve tri kategorije: dovoljno izražajan ulazni jezik poput logike višeg reda, ili teorije skupova koji bi omogućio konstrukciju modela na prirodan i koncizan način, automatska verifikacija bi trebalo da rastereti korisnika u što većoj meri prilikom izvođenja dokaza, i visoka garancija ispravnosti koja bi dala veliki stepen pouzdanja u dobijeni rezultat. U praksi, ovi ciljevi su u konfliktu. Interaktivni dokazivači enkodiraju bogatu logiku, koja je u osnovi vrlo izražajnih jezika za modelovanje. Njihovo okruženje za verifikaciju je obično izgrađeno nad relativno malim jezgrom za koje je na formalan način dokazana ispravnost, čime se osigurava da teoreme mogu biti izvedene isključivo iz eksplicitno navedenih aksioma i pravila. Sa druge strane, alati za automatsku verifikaciju poput SMT rešavača imaju fiksiran ulazni jezik za izražavanje modela, i implementiraju algoritme za automatsku verifikaciju za taj jezik. Kako bi dobro skalirali za veće probleme, koriste se sofisticirane optimizacije. Međutim, na taj način se nenamerno i lako može uvesti greška koja može uticati na ispravnost.

Očigledno je da bi bilo poželjno kombinovati interaktivne i automatske alate kako bismo prilikom verifikacije iskoristili prednosti oba alata. U te svrhe, interaktivni dokazivači često pružaju proširenja za alate za automatsku verifikaciju: dovoljno je izvršiti prevođenje formule koja se dokazuje u ulazni jezik alata za automatsko rezonovanje, i pokrenuti taj alat. Ako se dokaz uspešno izvede, dokazivač će formulu prihvatiti kao teoremu. Međutim, na ovaj način se i alat za automatsko rezonovanje uključuje u kod od poverenja (eng. *trusted code base*), i tako se oslabljuje garancija ispravnosti. Čak i da smo spremni da poverujemo u ispravnost alata za rezonovanje, funkcija za prevođenje formule iz logike višeg reda u logiku prvog reda je netrivialna, i time se stvara još jedna mogućnost za uvođenje grešaka.

Jedan način da se problemi izbegnu je da alat za rezonovanje proizvodi nekakav trag (eng. *proof traces*) kojim bi pokazao smer izvođenja dokaza, i koji kasnije može biti proveren nezavisno. Često je proveravanje ispravnosti samog dokaza mnogo jednostavnije od njegovog pronalaženja, tako da bi bilo prihvatljivo da alat za proveravanje postane deo koda od poverenja. Kombinacija alata pruža punu izražajnost interaktivnog dokazivača, uz automatizaciju alata za rezonovanje u njegovom domenu, bez smanjenja garancije ispravnosti.

Alternativan pristup bi mogao da bude verifikacija algoritma automatskog dokazivača unutar interaktivnog dokazivača teorema, i priključivanje koda čija je ispravnost dokazana kodu od poverenja. Tako bi se izbegla potreba za proveravanjem pojedinačnih dokaza. Ovakav pristup najverovatnije neće rezultovati dobrom implementacijom čija će efikasnost moći da se nadmeće sa dokazivačima implementiranim kao visoko-optimizovani C programi.

U radu je opisana implementacija provere dokaza u sklopu alata Isabelle, za procedure odlučivanja za logiku prvog reda bez kvantifikatora sa neinterpretiranim funkcijama i predikatskim simbolima implementiranim

u SMT rešavaču HarVey. Ovaj SMT rešavač kombinuje SAT rešavač sa procedurama za odlučivanje. Ukratko, SAT rešavač održava iskaznu apstrakciju ulazne formule. Iskazna apstrakcija se dobija tako što se atomi prvog reda zamenjuju iskaznim slovima. Kad god se pronađe model za ovu apstrakciju, posebna procedura odlučivanja proverava zadovoljivost dobijene konjunkcije baznih literala u teoriji. Ukoliko je pronađeni model nekompatibilan sa teorijom, proizvodi se konfliktna klauza kako bi se isključila jedna klasa modela. Ovaj proces se nastavlja sve dok se ne pronađe model, u kojem slučaju je ulazna formula zadovoljiva, ili dok SAT rešavač ne odluči da je iskazna apstrakcija nezadovoljiva.

2 Motivacija za integraciju alata

Motivacija za kombinovanje interaktivnih dokazivača i SMT rešavača se javila tokom verifikacije distribuiranih algoritama. Za verifikaciju je korišćen Isabelle, i taj formalizam je omogućio pisanje razumljivih sistemskih specifikacija. U nekoliko inicijalnih koraka je izvršeno instanciranje apstrakcija višeg reda, čime su dobijeni uslovi za verifikaciju formula prvog reda. Mnogi od dobijenih podciljeva su pripadali domenu automatskih procedura odlučivanja. Dokazivanje tih podciljeva je zahtevalo skup lema čije dokazivanje je više bilo dosadno nego teško. Ugrađeni mehanizam dokazivača za linearnu aritmetiku nije bio u stanju da dokaže leme automatski, čak ni nakon manuelnog instanciranja kvantifikatora. Za razliku od Isabelle, SMT rešavaču automatsko pronalaženje dokaza nije predstavljalo problem. Korišćenjem SMT rešavača kao proširenja interaktivnih dokazivača, korisnik može da se skoncentriše na verifikaciju na višem nivou, a da dosadne detalje ostavi alatu za automatsko rezonovanje.

3 Rekonstrukcija dokaza

SAT rešavači rešavaju problem zadovoljivosti za iskaznu logiku, i predstavljaju bitnu komponentu SMT rešavača. Za datu iskaznu formulu, SAT rešavač ili pronalazi zadovoljavajuću valuaciju, ili prijavljuje da formula nije zadovoljiva. Moderni SAT rešavači implementiraju DPLL algoritam, koji je unapređen mnogim optimizacijama poput analize konflikata, nehronoloških bektrekovanja, dobrih heuristika za grananja, i efikasnih struktura podataka. Ulaz za SAT rešavač je formula u konjunktivnoj normalnoj formi. Jednostavne konverzije formula u CNF dovode do problema, jer veličina rezultujuće formule može porasti eksponencijalno. S obzirom da za ovaj problem nije potrebno da rezultujuća formula u CNF bude ekvivalentna polaznoj, već je dovoljno da bude ekvizadovoljiva, za transformaciju se može iskoristiti Cajtinova transformacija nakon koje će veličina rezultujuće formule porasti linearno u odnosu na originalnu formulu.

Kod dokazivanja teorema, pokazujemo da je formula validna tako što pokušavamo da dokažemo da je njena negacija nezadovoljiva. SAT rešavači mogu da proizvedu niz koraka koji dovode do nezadovoljivosti, i to u vidu binarne rezolucije (svaki korak se može opisati sa dve klauze koje učestvuju u rezoluciji, i literalom na osnovu kojeg se vrši pravilo rezolucije). Ovaj izlaz iz SAT rešavača se prosleđuje Isabelle, gde se vrši provera dokaza nezadovoljivosti formule. Problem nezadovoljivosti formule se svodi na izvođenje kontradikcije iz klauza koje predstavljaju hipoteze.

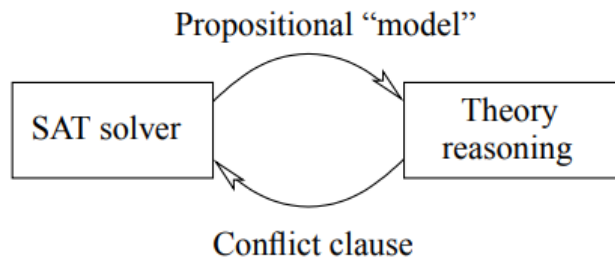
Na slici 1 su prikazani eksperimentalni rezultati za nekoliko različitih problema. Uspešno su rekonstruisani dokazi za probleme sa nekoliko sto-

tina klauza koji zahtevaju oko 10000 binarnih rezolucija. 'SAT time' predstavlja vreme izvršavanja SAT rešavača, dok 'Total time' uključuje vreme koje je bilo potrebno Isabelle da rekonstruiše dokaz. Može se primetiti da je vreme potrebno za samu rekonstrukciju višestruko veće od vremena potrebnog za pronalaženje dokaza, iako je problem pronalaženja dokaza teži. Pretpostavlja se da je glavni uzročnik ovoga složena interna reprezentacija formula i teorema u interaktivnom dokazivaču koje su prilagođene logici višeg reda, i nisu optimizovane za iskaznu logiku.

Problem	# clauses	SAT time	Total time
MSC007-1.008	204	0.208	11.546
PUZ015-2.006	184	0.005	2.435
PUZ016-2.005	117	0.003	1.158
PUZ030-2	63	0.002	0.485
PUZ033-1	13	0.003	0.078
SYN090-1.008	65	0.002	0.492
SYN093-1.002	26	0.005	0.133
SYN094-1.005	82	0.005	0.742

Slika 1: Vreme izvršavanja rekonstrukcije dokaza dobijenih SAT rešavačem

Integracija SAT rešavača sa Isabelle je ključna za podršku SMT rešavačima koji obrađuju izražajnije jezike bez kvantifikatora. SMT rešavači su SAT rešavači koji rade zajedno sa procedurama odlučivanja za određene teorije, kao što je prikazano na slici 2. Informacije koje se razmenjuju na interfejsu su konfliktne klauze dobijene u proceduri odlučivanja. Konfliktne klauze se izvode zakonima jednakosne logike (refleksivnost, simetričnost, tranzitivnost i kongruentnost).



Slika 2: Saradnja između SAT rešavača i procedure odlučivanja

Algoritam za kongruentno zatvorenje određuje zadovoljivost skupa literala logike prvog reda u teoriji sa neinterpretiranim funkcijama i predikatima, i to konstrukcijom klasa ekvivalencije termova. Postoje mnoge implementacije algoritama kongruentnih zatvorenja, naročito Nelson-Openovog algoritma. Složenost tog algoritma je kvadratna. Postoje i efikasniji algoritmi koji postižu asimptotsku složenost $O(n \log n)$.

U nastavku je opisana implementacija interfejsa između Isabelle i haR-Vey. Glavna ideja je da se SAT rešavač ne iskoristi za davanje kompletnog dokaza za dati cilj, već se koristi za generisanje niza lema (konfliktnih klauza), koje će usmeravati Isabelle prilikom pronalaženja dokaza. Za datu formulu F , pomenuti interfejs obavlja sledeće korake:

1. Negacija formule F se konvertuje u SMT-LIB format i prosleđuje se SMT rešavaču.
2. Ako je $\neg F$ nezadovoljiva, SMT rešavač proizvodi niz formula C_1, \dots, C_n (konfliktnih klauza), zajedno sa tragovima dokaza za svaku od klauza. Ako je formula zadovoljiva, interfejs prikazuje model koji je pronađen od strane SMT rešavača, i izvršavanje se prekida.
3. U Isabelle se konstruiše dokaz za svaku od konfliktnih klauza C_i , a pronalaženje dokaza je vođeno tragovima proizvedenim od strane SMT rešavača.
4. Konstruiše se dokaz za $[\neg F; C_1; \dots; C_n] \Rightarrow False$.
5. Primenjuje se modus ponens nad formulama dobijenim u prethodna dva koraka kako bismo došli do $\neg F \Rightarrow False$, i na taj način se dokazuje F .

4 Zaključak

U radu je predložena tehnika za kombinovanje interaktivnih dokazivača teorema i SMT rešavača. Pošto su dokazi sertifikovani od strane formalno verifikovanog kernela interaktivnog dokazivača, teoreme dokazane na ovaj način imaju isti nivo garancije ispravnosti kao i teoreme dokazane isključivo interaktivno. Kombinovanje sa eksternim alatom za rezonovanje omogućava značajno podizanje nivoa automatizacije, uz zadržavanje izražajnosti ulaznog jezika interaktivnog dokazivača.

Literatura

- [1] Pascal Fontaine et al. Expressiveness + Automation + Soundness: Towards Combining SMT Solvers and Interactive Proof Assistants. *TACAS'06 Proceedings of the 12th international conference on Tools and Algorithms for the Construction and Analysis of Systems*, pages 167–181, 2006.