Matematički fakultet u Beogradu

Seminarski rad iz predmeta automatsko rezonovanje

Binarni dijagrami odlučivanja - BDD

Studenti:
Nikola Stanojević,
1064/2012
Miloš Milaković,
1063/2012

Profesor:
Filip Marić

July 9, 2013

Sadržaj

1 Uvod

Binarni dijagram odlučivanja (BDD) je direktan acikličan graf koji se koristi za predstavljanje diskretnih funkcija. Svaki nezavršni čvor ima tačno dva čvora potomka (stepen izlaznog grananja 2) dok dva završna čvora predstavljaju konstantnu funkciju 1, odnosno konstantnu funkciju 0. Značajna osobina ovih grafova je kanoničnost, odnosno za dve Bulove funkcije možemo da tvrdimo da su jednake ukoliko su njihove BDD-reprezentacije jednake.

Druga bitna osobina ove strukture podataka je da su rezultati obrade na raspolaganju za dalju upotrebu. BDD se formira eliminacijom dva tipa redundantnosti. Transformacije kojima se eliminišu redundantnosti su:

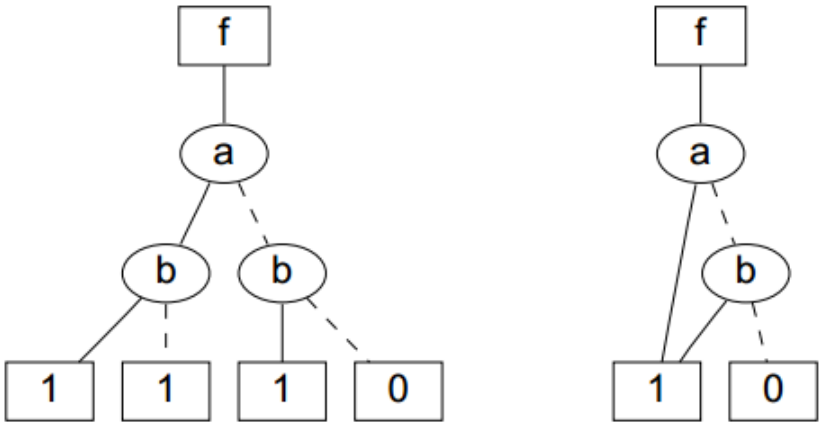
(1) eliminacija čvorova dijagrama čije su obe izlazne grane usmerene ka istom čvoru
(2) medjusobno izomorfni podgrafovi se spajaju u jedan podgraf

Varijante BDD koje su našle primenu u kriptografiji su Uredjen binarni dijagram odlučivanja (Ordered Binary Decision Diagram, OBDD), Slobodni binarni dijagram odlučivanja (Free Binary Decision Diagram, FBDD), kao i Binarni dijagram odlučivanja sa potisnutim nulama (Zero-suppressed Binary Decision Diagram, ZBDD).

FBDD je BDD u kojem se duž svakog puta promenjive pojavljuju najviše jedanput. OBDD je dijagram odlučivanja u kojem je pored uslova da se svaka promenjiva pojavljuje isključivo jedanput duž svakog puta i da je redosled promenjivih isti duž svakog puta. Redukovani BDD (ROBDD) je OBDD koji je redukovan sa dva pravila redukcije: pravilo brisanja i pravilo spajanja. Ovim pravilima redukcije uklanjaju se redundantnosti iz OBDD dijagrama. Binarni dijaram odlučivanja sa potisnutim nulama (ZBDD) daje jedinstvenu i kompaktnu reprezentaciju skupova. Ovom strukturom podataka je manipulisanje skupovima mnogo jednostavnije i efikasnije u odnosu na originalan BDD.

2 Od binarnog stabla odlučivanja do binarnog dijagrama odlučivanja

Binarni dijagrami odlučivanja su umanjena binarna stabla odlučivanja.



Slika 1: Binarno stablo odlučivanja i binarni dijagram odlučivanja za disjunkciju a i b.

Na levoj strani slike 1 prikazano je binarno drvo odlučivanja za disjunkciju promenljivih a i b. Početni čvor (čvor na vrhu obeležen sa f) je čvor funkcije.

Čvorovi u obliku elipse obeleženi nazivom promenljivih su unutrašnji čvorovi, a čvorovi u kvadratima na dnu su listovi datog binarnog stabla odlučivanja. Listovi su obeleženi ili sa 1 (predstavlja tačno) ili sa 0 (predstavlja netačno).

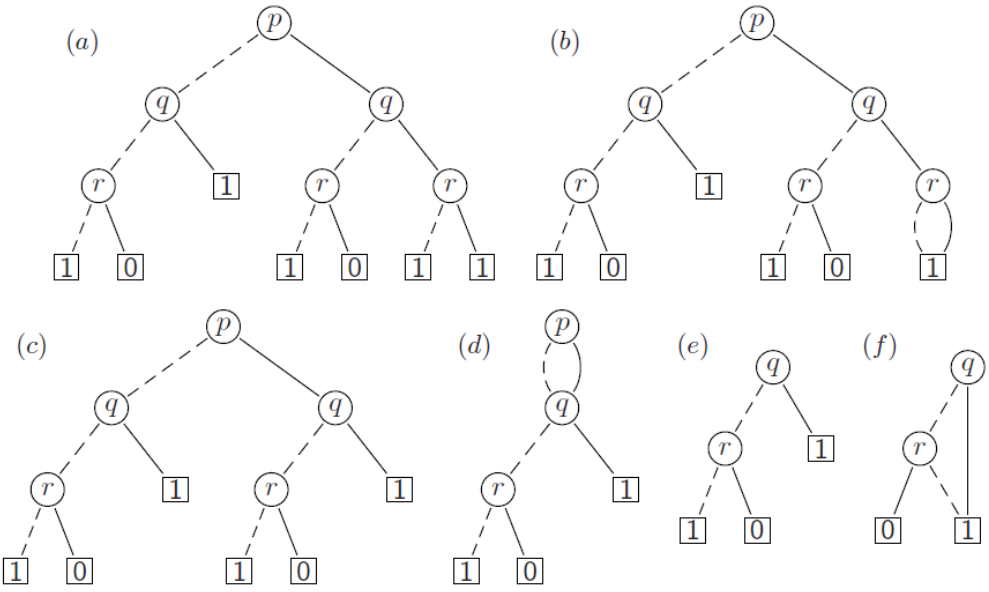
Vrednost funkcije f za datu valuaciju v vrednosti promenljivih a i b zavisi od putanje od početnog čvora do nekog od listova. Vrednost čvora u listu predstavlja vrednost funkcije f u toj valuaciji v. Za svaki unutrašnji čvor krećemo se putanjom koja je označena punom linijom ukoliko je za vrednost promenljive kojom je taj čvor označen postavljeno 1, ili isprekidanom linijom ukoliko je za vrednost promenljive kojom je taj čvor označen postavljeno 0.

Grane označene punom linijom nazivaju se *then* grane, a grane označene isprekidanom linijom nazivamo *else* granama.

Na desnoj strani slike 1 prikazan je binarni dijagram odlučivanja. Dobijen je od odgovarajućeg binarnog stabla odlučivanja procesom koji se zove *redukcija*.

Redukcija se zasniva na upotrebi naredna dva pravila na stablo odlučivanja sve dok se bilo koje od njih moze primeniti:

- ako su dva čvora listovi i imaju iste vrednosti, ili su unutrašnji i imaju iste potomke, tada se ta dva čvora spajaju u jedan.
- ako neki od unutrašnjih čvorova ima iste potomke tada se on uklanja iz dijagrama i prethodni čvor se spaja sa njegovim potomkom.



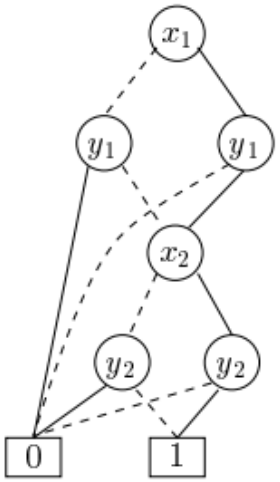
Slika 2: Prikaz procesa redukcije (svodjenja stabla odlučivanja na binarni dijagram odlučivanja)

3 Uredjeni dijagram binarnog odlučivanja - OBDD

Kada binarnom dijagramu odlučivanja dodamo svojstvo uredjenosti dobijamo uredjeni dijagram binarnog odlučivanja.

Uredjenost predstavlja jedno od ključnih ograničenja koje je omogućilo algoritamsku formulaciju. Uredjenost podrazumeva da se promenljive duž svakog puta od čvora do lista moraju pojavljivati u fiksnom rasporedu. Binarni dijagram odlučivanja je uredjen ako je zadato potpuno uredjenje $<$ nad skupom promenljivih, pa važi: za bilo koji čvor u i za svakog njegovog potomka v koji nije list, njihove pripadajuće promenljive moraju imati uredjenje $var(u) < var(v)$.

Odabir uredjenosti promenljivih značajno utiče na složenost uredjenog binarnog stabla odlučivanja. Zato je veoma značajno odaberati uredjenost promenljivih koja će doprineti da dobijemo najmanju moguću složenost uredjenog binarnog stabla odlučivanja.



Slika 3: Primer uredjenog dijagrama binarnog odlučivanja za formulu $(x_1 \Leftrightarrow y_1) \wedge (x_2 \Leftrightarrow y_2)$ sa uredjenjem $x_1 < y_1 < x_2 < y_2$.

3.1 Odabir uredjenja promenljivih

Veličina a samim tim i složenost uredjenog binarnog stabla odlučivanja u mnogome zavisi od funkcije koju predstavljamo koristeći OBDD ali i od odabranog uredjenja promenljivih.

Postoje funkcije $f(x_1, ..., x_n)$ za koje pri odabiru dobre strategije uredjenja promenljivih dobijamo linearan broj čvorova prilikom pravljenja OBDD, a pri odabiru loše strategije uredjenja promenljivih moguće je čak dobiti i eksponencijalan broj čvorova prilikom pravljenja OBDD.

Ako posmatramo funkciju $f(x_1, ..., x_n) = x_1x_2 + x_3x_4 + \cdots + x_{2n-1}x_{2n}$ sa sledećim uredjenjem promenljivih:

$$x_1 < x_3 < \cdots < x_{2n-1} < x_2 < x_4 < \cdots < x_{2n}$$

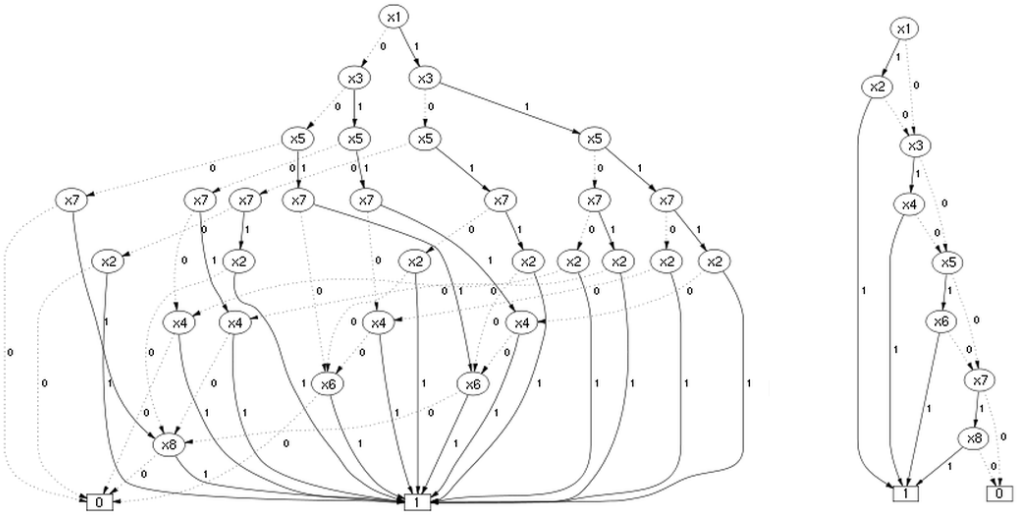
tada je potrebno 2^{n+1} čvorova za predstavljanje funkcije u OBDD što je eksponencijalne složenosti. Medjutim ako za uredjenje promenjljivih odaberemo:

$$x_1 < x_2 < x_3 < x_4 < \cdots < x_{2n-1} < x_{2n}$$

tada nam je potrebno samo $2n+2$ čvorova za predstavljanje funkcije u OBDD pa dobijamo linearnu složenost.

Iz prethodnog primera vidimo da je odabir uredjenja promenljivih od velike važnosti prilikom upotrebe ove strukture podataka u praksi. Problem pronalaženja najboljeg uredjenja promenljivih za neku funkciju je NP-težak problem.

Medjutim postoje i funkcije za koje je složenost uvek eksponencijalna, bez obzira na odabir uredjenja promenljivih. Ovo važi za multiplikativne funkcije usled kompleksnosti faktorizacije.



Slika 4: OBDD za funkciju $f(x_1, ..., x_n) = x_1x_2 + x_3x_4 + x_5x_6 + x_7x_8$ sa korišćenjem lošeg (slika levo) i dobrog (slika desno) uredjenja promenljivih.

4 Redukovani uredjeni dijagram binarnog odlučivanja - ROBDD

4.1 Redukcija

Ako pored svojstva uredjenosti uvedemo i svojstvo redukovanosti dobijamo redukovani uredjeni dijagram binarnog odlučivanja.

Ovakav binarni dijagram odlučivanja predstavlja prikladnu strukturu podataka za simboličku manipulaciju bulovih funkcija. ROBDD je jedan od najčešce korišćenih oblika binarnog drveta odlučivanja.

Dva pravila transformacije kojima se postiže redukovanost (pravila redukcije), gde funkcija $low()$ predstavlja 0 grane, a funkcija $high()$ predstavlja 1 grane:

- *Uklanjanje višestrukih nezavršnih čvorova:* Ako za dva nezavršna čvora u i v važi $var(u) = var(v)$, $low(u) = low(v)$ i $high(u) = high(v)$ tada se jedan od ta dva čvora uklanja i sve dolazne grane se preusmere u preostali čvor.

- *Uklanjanje nepotrebnih nezavršnih čvorova:* Ako za nezavršni čvor v važi $low(v) = high(v)$, tada se čvor v uklanja i sve dolazne grane se preusmeravaju u $low(v)$.

Posledica:
Dve logičke funkcije su ekvivalentne ako su njihovi ROBDD grafovi izomorfni. *Izomorfnost* je postignuta ako postoji bijekcija izmedju grafova takva da se završni čvorovi preslikavaju u završne, a nezavršni u nezavršne s istim vrednostima dece.

Uslov je jednaka uredjenost promenljivih. Time se bitno pojednostavljuje odlučivanje o ekvivalentnosti logičkih formula i zadovoljivosti (koja se svodi na pokazivanje ne-ekvivalencije s grafom funkcije $f = 0$).

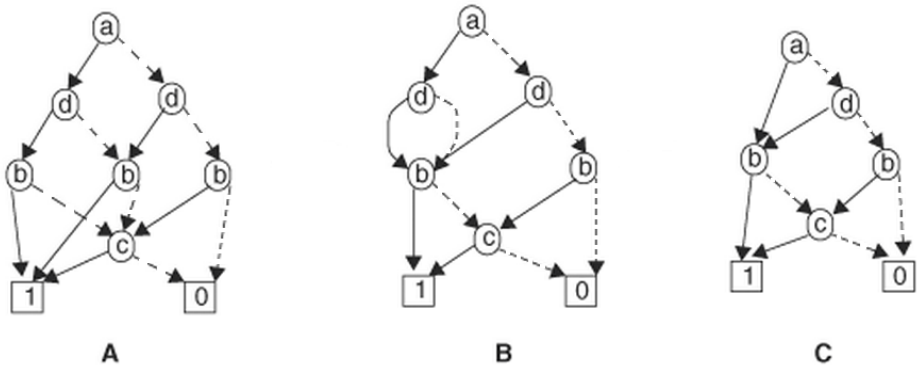
Nedostaci ROBDD:

- Za mnoge logičke funkcije veličina ROBDD-a (broj čvorova) je polinomijalna u odnosu na broj promenljivih ali **samo za dobru uredjenost promenljivih**.
- Za neke logičke funkcije veličina ROBDD-a je eksponencijalna u odnosu na broj promenljivih **bez obzira na uredjenost**.
- Ozbiljni i složeni realni sistemi moraju pronaći dobru uredjenost promenljivih (optimalna uredjenost je NP teška).
- Rešenje uredjenost:

  Statičko (zasnovano na topologiji sistema)

  Dinamičko (eksperimentisati s uredjenošću)

  Korisna heuristika: uzmi promenljivu koja je najzastupljenija u logičkom izrazu (nalazi se u najvećem broju produktnih članova.
  Npr.: $f = (ac + cd)$, dobra uredjenost: (c, a, d)



Slika 5: Operacija redukcije nad OBDD: (A) Originalni OBDD; (B) Spajanje dva b čvora; (C) Eliminisanje čvora d.

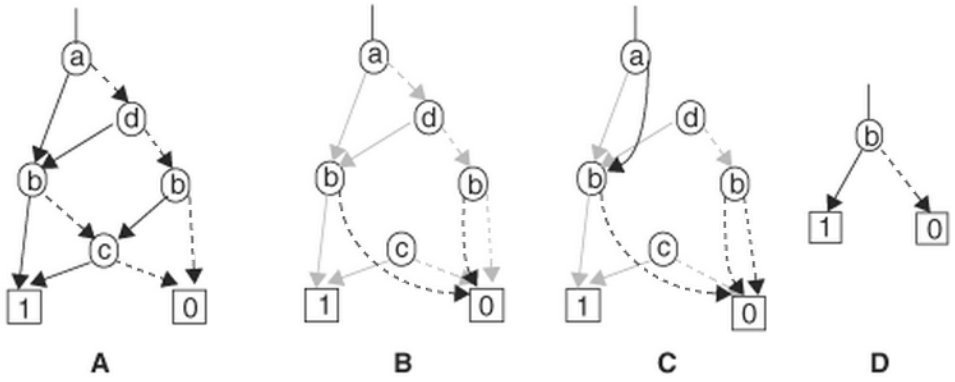
4.2 Restrikcija

Pored operacije redukcije nad OBDD se može vršiti i operacija restrikcije. Operacija restrikcije primenjena na neku funkciju postavlja odabrane promenljive na odredjene vrednosti (0 ili 1).

Ako posmatramo funkciju $f(r, s, x, y, z) = sr\overline{x}z + \overline{rs}y + \overline{rsx}y + \overline{x}y\overline{z}$, primenom restrikcije na ovu funkciju postavljanjem vrednosti x na 0 i y na 1 dobijamo funkciju $f(r, s, z) = srz + \overline{rs} + \overline{z}$. Ova operacija se može jednostavno obaviti korišćenjem binarnih dijagrama odlučivanja. Ako se vrši restrikcija tako što se promenljiva p postavlja na vrednost 1, potrebno je jednostavno preusmeriti sve grane od čvorova koje dolaze u čvor p ka njihovim čvorovima koji su obeleženi 1-granama. Slično tome je i postavljanje promenljive p na vrednost 0 korišćenjem operacije restrikcije.

Moguće je da se nakon preusmeravanja grana pojave redundantni i ekvivalentni čvorovi, pozivom operacije redukcije dijagram ćemo dovesti do ROBDD.

Takodje čvorovi označeni sa p će i dalje postojati u binarnom stablu odlučivanja iako nemaju dolaznih grana, pa ih je potrebno ukloniti iz binarnog dijagrama odlučivanja. Njihovo brisanje postižemo tako što obrišemo sve čvorove, osim korenog čvora, koji nemaju dolazne grane.



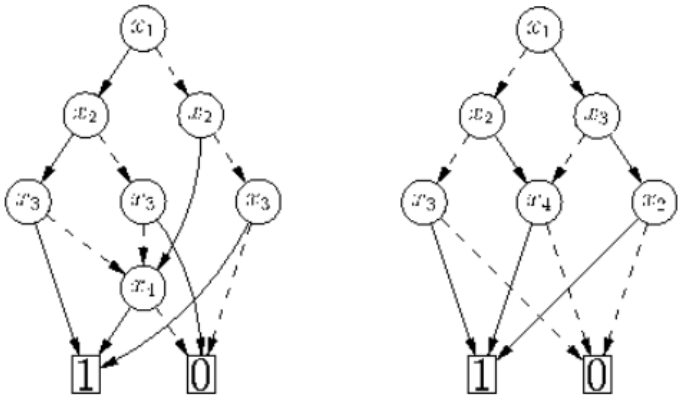
Slika 6: Operacija restrikcije nad OBDD: (A) Originalni OBDD; (B) Restrikcija za $c = 0$; (C) Restrikcija za $d = 1$; (D) RBDD.

5 Slobodni binarni dijagram odlučivanja - FBDD

Slobodni binarni dijagram odlučivanja ima dodatni uslov u odnosu na osnovni BDD. Taj uslov je da je svaka promenljiva testirana najviše jednom prilikom izračunavanja. Predstavio ga je Masek 1976. godine i on ih je tada nazvao granajućim programima (en. Branching programs).

Za razliku od OBDD-a dozvoljava da promenljive budu različito rasporedjene duž različitih putanja. Takodje za razliku od OBDD-a koji može da zahteva i eksponencijalni broj čvorova, FBDD zbog svojih karakteristika zahteva polinomijalni prostor za predstavljanje. Glavni problem ovih dijagrama je nepostojanje efikasne heuristike za njihovo konstruisanje.
Značajne osobine FBDD grafova su da oni mogu efikasno da se minimizuju i da omogućavaju efikasno izlistavanje.

Slika 7 prikazuje minimalni OBDD i FBDD funkcije $f(\overline{x_1}\overline{x_2}x_3 + \overline{x_1}x_2x_4 + x_1\overline{x_3}x_4 + x_1x_2x_3)$. Pošto OBDD mora ispoštovati fiksni raspored promenljivih za svaku putanju, za njegovu reprezentaciju je potrebno sedam čvorova, dok je za FBDD reprezentaciju dovoljno šest čvorova.



Slika 7: Prikaz dijagrama za funkciju $f(\overline{x_1}\overline{x_2}x_3 + \overline{x_1}x_2x_4 + x_1\overline{x_3}x_4 + x_1x_2x_3)$: (A) Minimalni OBDD; (B) Minimalni FBDD.

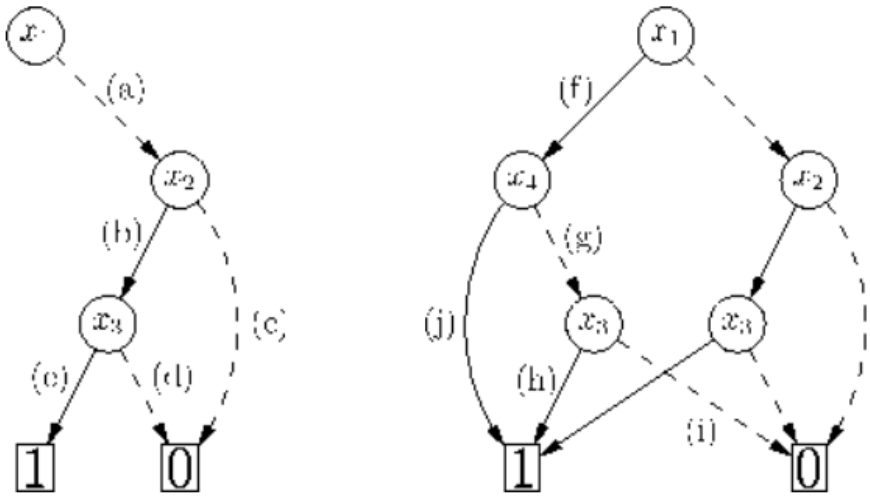
5.1 Implementacija FBDD-a korišćenjem SAT rešavača

Pretpostavka: Neka je $f : B^n \to B$ buleanska funkcija u KNF normalnoj formi. Postoje tri svojstva konstrukcije FBDD dijagrama:

- Svaki zadovoljavajući zadatak pronadjen od strane SAT rešavača se odnosi na *1-putanju* FBDD-a koji predstavlja funkciju f.
- Svaki konfliktni zadatak prepoznat od strane SAT rešavača se odnosi na *0-putanju* FBDD-a koji predstavlja funkciju f.
- Svaka implikacija za $x_i = b, (x_i \in f, b \in B)$ izvršena od strane SAT rešavača vodi ka *0-putanji* FBDD-a kojim je predstavljena funkcija f. Ova putanja može biti konstruisana korišćenjem trenutnog zadatka SAT rešavača i korišćenjem $x_i = b$.

Prema ovim svojstvima *snimanje* pojedinačnih koraka tokom procesa pretrage koju vrši SAT rešavač vodi ka parcijalnom FBDD-u.

5.2 Primer implementacije FBDD-a korišćenjem SAT rešavača



Slika 8: Korišćenje SAT rešavača za implementiranje FBDD-a.

Konstruišemo FBDD za funkciju $f(x_1, x_2, x_3, x_4) = (x_1+x_2)(x_3+x_4)(x_1+x_3+\overline{x_4})$. U ovom primeru smo *pamtili* svaki korak SAT rešavača. Na slici gore su sa leve strane prikazani koraci istog.

Na početku SAT rešavač dodeljuje $x_1 = 0$ koristeći heuristiku odlučivanja (a). Zbog prve klauze imamo da je $x_2 = 1, (b)$. Kako je poslednji korak implikacija, prema pravilu 3, zaključujemo da su $x_1 = 0$ i $x_2 = 0$ konfliktna dodeljivanja (npr. $f(0, 0, x_3, x_4) = 0$ je predstavljeno *0-putanjom*, (c)). U sledećem koraku vrši se dodeljivanje $x_3 = 0$. Ovo izaziva konflikt i *0-putanja* je zapamćena (d) prema pravilu 2.

Nakon povratka (en. backtracking) do $x_3 = 1$, sve klauze postaju zadovoljive i pronadjeno je zadovoljivo dodeljivanje. Ovo je zapamćeno *1-putanjom* (e) prema pravilu 1. Pošto je pronadjeno zadovoljivo dodeljivanje, SAT rešavač prestaje sa radom.

6 Binarni dijagram odlučivanja sa potisnutim nulama - ZBDD, ZSDD

Binarni dijagram odlučivanja sa potisnutim nulama je jedna vrsta binarnih dijagrama odlučivanja kod kojeg se čvorovi ne predstavljaju kada su im pozitivni i negativni deo različiti nego onda kada je negativni deo različit od konstante 0. Predstavio ga je Šin-Iči Minato 1993. godine, oko 7 godina nakon predstavljanja osnovnih ideja o BDD strukturi podataka od strane Randala Brajanta.

Definisan je kao OBDD, ali je pravilo eliminacije prilagodjeno tako da eliminiše proredjenost.

Definicija: Definisemo dva redukciona pravila:

Pravilo eliminacije: Ako jedna grana čvora v pokazuje na *0-list*, onda elimišemo čvor v i preusmeravamo sve ulazne grane čvora v *0-nasledniku* cvora v.

Pravilo spajanja: Ako su unutrašnji čvorovi u i v označeni od strane iste promenljive, njihove *1-grane* vode ka istom čvoru i njihove *0-grane* vode ka istom čvoru onda eliminišemo jedan od čvorova u ili v i preusmeravamo sve dolazne grane obrisanog čvora u preostali.
Ovi dijagrami su naročito korisni kada su primenjeni na funkcije koje u skoro svim slučajevima vraćaju nulu.

Donald Knut je 2011. u razgovoru za *"All Questions Answered"* izjavio da su binarni dijagrami odlučivanja sa potisnutim nulama najlepša konstrukcija u računarstvu. Dok BDD predstavlja strukturu podataka za buleanske funkcije, ZBDD je sa druge strane struktura podataka za kako ih je Knut nazvao porodice skupova.

Porodice skupova predstavljaju još jedan način posmatranja buleanskih funkcija , kako postoji prirodna korespodencija izmedju rešenja buleanskih funkcija i porodica skupova.

Svako rešenje funkcije korespondira sa odredjenim podskupom. Iako se

porodice skupova mogu enkodirati kao buleanske funkcije, ponekad ih je bolje shvatati kao porodice skupova nego u buleanskom smislu.

Porodice podskupova su uzete iz dobro uredjenog univerzuma U. Koristimo sledeća pravila za predstavljanje ZBDD-a:

- Prazna porodica (en. *Empty Family*) - $\emptyset$ se predstavlja znakom $\perp$
- Jedinična porodica (en. *Unit Family*) - $\{\emptyset\} = \epsilon$, se predstavlja znakom $\top$
- Ako je f bilo koja druga porodica, znamo da nije prazna, i da najmanje jedan njen član nije prazan. U tom slučaju neka je v poslednji element univerzuma U kojeg f podržava. To v se pojavljuje u najmanje jednom skupu u f. Tada definišemo podporodice f_0 i f_1 porodice f sa: $f_0 = \{\alpha \mid \alpha \in f \wedge v \notin \alpha\}$ i $f_1 = \{\alpha \mid \alpha \cup \{v\} \in f \wedge v \notin \alpha\}$.

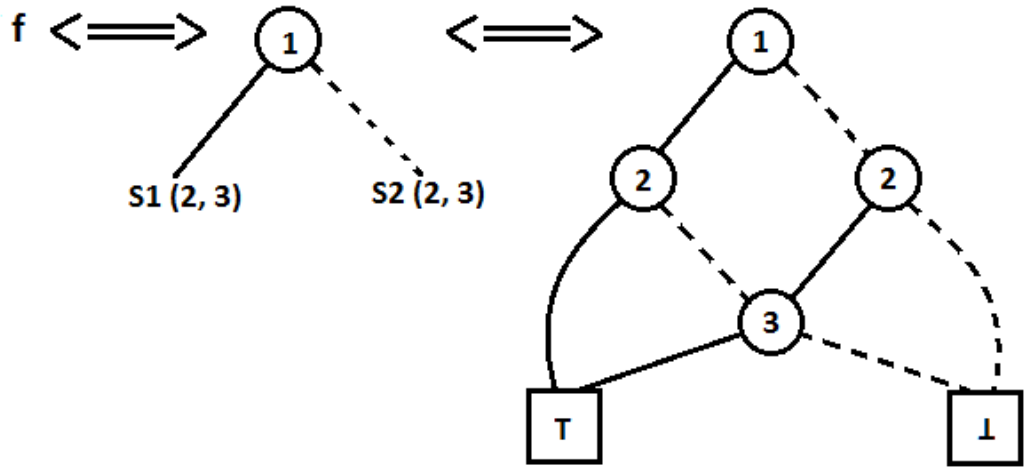
  Ove podporodice odgovaraju onim skupovima koji sadrže i ne sadrže v.

Primer:

Neka je f porodica dvočlanih podskupova skupa: $\{1, 2, 3\}$. $f = S2(x1, x2, x3)$. Porodica f je $\{\{1, 2\}, \{1, 2\}, \{2, 3\}\}$ pa je njena najmanja podrška 1.

Ispostavlja se da su podporodice f_0 i f_1 redom: $S_2(2, 3) = \{\{2, 3\}\}$ i $S_1(2, 3) = \{\{2\}, \{3\}\}$. Tada imamo *low()* i *high()* grane koje idu od 1 do ZBDD-a od podporodica f_0 i f_1 koje rekurzivno konstruišemo.

Dostupni su paketi za programske jezike C i JAVA pod nazivima CUDD i JDD koji implementiraju osnovne BDD i ZDD operacije.



Slika 9: f porodica dvočlanih podskupova skupa $\{1, 2, 3\}$.

7 Zaključak

Binarni dijagrami odlučivanja su popularna struktura podataka koja se koristi kod mnogih verifikacionih algoritama, u istrazivanju podataka, kriptografiji... Ipak smatramo da im nije posvećena dovoljna pažnja iako predstavljaju jedan od najlepših oblika struktura podataka. U ovom radu smo predstavili četiri najkorišćenije vrste binarnih dijagrama odlučivanja.

8 Reference

- An Introduction to Binary Decision Diagrams - Henrik Reif Andersen:
  http://www.cs.unb.ca/~gdueck/courses/cs4835/bdd97.pdf
- Binary Decision Diagrams - Fabio Somenzi:
  http://www.ecs.umass.edu/ece/labs/vlsicad/ece667/reading/somenzi99bdd.pdf
- Wikipedia - Slobodna enciklopedija:
  http://en.wikipedia.org/wiki/OBDD
  http://en.wikipedia.org/wiki/Zero-suppressed_decision_diagram
- Building Free Binary Decision Diagrams Using SAT Solvers:
  http://facta.junis.ni.ac.rs/eae/fu2k73/7wille.pdf
- An Introduction to Zero-Suppressed Binary Decision Diagrams
  http://www.eecs.berkeley.edu/~alanmi/publications/2001/tech01_zdd.pdf
- Fun with ZDDs
  http://ashutoshmehra.net/blog/2008/12/notes-on-zdds/