Uniformno svodjenje teških problema na SAT

Predrag Janičić
URL: www.matf.bg.ac.yu/~janicic
Matematički fakultet, Univerzitet u Beogradu, Srbija

ARGO seminar
Beograd, 25.02.2009.

Plan izlaganja

- Početne ideje
- Uniformno svodjenje kriptoanalitičkih problema na SAT
- Uniformno svodjenje problema na SAT
- Dalji rad

Početne ideje

- Nisu sve SAT instance jednako teške za rešavanje
- Postoji region fazne promene koji razdvaja dominantno zadovoljive od dominantno nezadovoljivih formula i u kojem su formule najteže
- Pitanje: ako se neki težak problem (npr. kriptoanalitički) svede na SAT, da li će odgovarajuće SAT instance da budu teške?
- Pitanje: da li boljim kriptografskim algoritmima odgovaraju teže SAT formule?

Svodjenje heš funkcija na SAT

- Motivisano gornjim pitanjima
- Zajednički rad sa Dejanom Jovanovićem (2004/05)
- Analizirane funkcije MD4 i MD5
- Pokazalo se da je svodjenje na SAT na osnovu specifikacije nepraktično ili neizvodivo

Osnovne ideje metoda

- Metod za generisanje SAT formula koje odgovaraju kriptoanalitičkom problemu na osnovu:
  - implementacije kriptografskog algoritma u jeziku C/C++
  - korišćenja polimorfizma jezika C/C++

Polimorfizam

- Svojstvo programskog jezika koje omogućava različite obrade različitih tipova podataka kroz jedinstveni interfejs
- Preopterećivanje operatora je specifična forma polimorfizma, u kojoj se standardni operatori tretiraju kao polimorfne funkcije

Implementacija

- Autor implementacije: Milan Šešum (2008)
- `Formula` — osnovna klasa za sve formule
- `SequenceOfFormulae` — klasa koja preuzima ulogu celobrojnih tipova podataka, koji su zamenjeni nizovima formula; preopterećeni su (logički i aritmetički) operatori koji se koriste u implementaciji kriptografskih funkcija
- Generisanje formule direktno prati izvršavanje, pa izvršavanje mora da bude isto za sve ulaze; to znači da petlje i grananja ne mogu da zavise od ulaznih vrednosti
- Zadata vrednost se uparuje sa dobijenim izlazom i to daje traženu SAT formulu

Implementacija (nastavak)

- Primer: preopterećivanje operatora ~

```
SequenceOfFormulae SequenceOfFormulae::operator ~ () {
   SequenceOfFormulae not;

   /* Compute NOT */
   for (int i = 0; i < bitArray.size(); i++) {
      Formula *f = new FormulaNot(bitArray[i]);
      not.setFormulaAt(i, f);
   }

   return not;
}
```

Dobijeni rezultati

- Pristup je primenjen na
  - kriptoanalizu heš funkcija MD4 i MD5 (2005)
  - kriptoanalizu simetričnog algoritma DES (2008)
- Rešene su samo oslabljene verzije problema — za manji broj ulaznih bitova ili za nekoliko rundi
- Smatra se jednim od prvih pristupa za logičku kriptoanalizu
- Zhang i Mironov u jednom radu (2007) komentarišu pristup i sa pravom konstatuju da je neophodno kombinovanje sredstava klasične i logičke kriptoanalize

Mogućnosti pristupa

Uprkos nedostacima, pristup ima i prednosti:

- Svodjenje na SAT je uniformno; npr. dok je Massacciju (1999) za prevodjenje DES-a na SAT bilo potrebno verovatno nekoliko nedelja, sa ovim pristupom dovoljno je nekoliko minuta
- Moguće je generisanje zadovoljivih i nezadovoljivih SAT formula čija se težina skalabilna, a to je jedan od najvećih modernih izazova u SAT svetu
- Pristup se može primeniti i na potpuno druge domene

Osnovna ideja

- Kod kriptografskih algoritama, treba naći ključ k koji daje neki šifrat s od poruke m:

$$C(k, m) = s$$

ili naći poruku m koja daje zadatu vrednost h heš funkcije:

$$hash(m) = h$$

- U oba slučaja, traži se inverzna vrednost funkcije
- Tražena vrednost može biti reprezentovana nizom bitova i nadjena grubom silom, ali je poželjno efikasnije rešenje
- Bolje je pametno problem svesti na SAT, generisanjem formula koje odgovaraju izračunavanjima

Domen

- Kriptografske funkcije
- Izračunavanje celobrojnog korena
- Pitanje: koji se sve problemi mogu rešavati na ovaj način (invertovanjem)?

Domen (nastavak)

- Odgovor: mogu se, izmedju ostalog, rešavati problemi odlučivanja
- Za problem odlučivanja, provera da li je x svedok opisuje se obično nekom jednostavnom funkcijom f; odgovor na problem je pozitivan ako i samo ako postoji x takav da važi $f(x) = 1$
- Dakle, rešavanje problema odlučivanja se svodi na invertovanje funkcije provere

Primeri: sudoku

- Da li postoji sudoku sa zadatim fiksnim elementima?
- Ukoliko neko ponudi rešenje — lako ga je proveriti
- Potencijalno rešenje se reprezentuje kao niz bitova a proverava jednostavnom funkcijom f
- Rešenje x zadovoljava uslov $f(x) = 1$, a odgovor na pitanje je pozitivan akko je zadovoljiva formula dobijena invertovanjem funkcije f; sâmo rešenje se rekonstruiše iz dobijene valuacije

Primeri: problem klika

- Da li postoji potpuni podgraf veličine k datog grafa?
- Za svaki podgraf lako je proveriti da li predstavlja potpuni podgraf veličine k
- Uniformnim invertovanjem funkcije provere, dolazi se do SAT formule koja je zadovoljiva ako i samo ako je odgovor problema pozitivan

Primeri: SAT

- Da li postoji zadovoljavajuća valuacija za datu iskaznu formulu?
- Za svaku valuaciju lako je proveriti da li je zadovoljavajuća
- Uniformnim invertovanjem funkcije provere, dolazi se do SAT formule koja je zadovoljiva ako i samo ako je odgovor problema pozitivan
- Da li je ovo svodjenje problema na samog sebe nepotrebno?

Primeri: NP-kompletni problemi

- Svi su problemi odlučivanja
- Provera da li je nešto rešenje problema je uvek efikasna (tj. polinomijalna)
- Uniformnim invertovanjem funkcije provere, dolazi se do SAT formule koja je zadovoljiva ako i samo ako je odgovor problema pozitivan
- Ogroman skup važnih problema kao domen metoda

Potencijalni problemi i dileme

- Prilikom konstruisanja SAT formula koristi se tzv. definiciona forma koja uvodi ogroman broj novih promenljivih i klauza
- Bolja interna reprezentacija bi mogla da se isplati?
- Ili ne — da li se, bez obzira na pristup kodiranju u SAT, dobijaju formule približno iste težine?

Potencijalni problemi i dileme (nastavak)

- Zbog ograničenja na istovetan tok izvršavanja programa, mora se napraviti poseban program za svaki npr. ulazni graf
- U domenu SAT rešavanja, poželjno je prilagoditi SAT rešavač, verovatno tako da za *decide* promenljive uzima osnovne promenljive, tj. promenljive koje reprezentuju potencijalno rešenje

Dalji rad

- Zajednički rad sa Milanom Šešumom
- Duboka interakcija sa SAT rešavačem
- Dobre specifikacije i dobar jezik specifikovanja (neće prihvatati konstrukcije koje metod ne može da podrži)
- Obimna poredjenja sa specijalizovanim rešenjima za razne probleme
- Automatsko generisanje invarijanti, veza sa statičkom analizom (Viktorova ideja)

Zaključci

- Moguće je uniformno svodjenje teških problema na SAt
- Efikasna rešenja zahtevaju dalja unapredjenja trenutne metodologije