Novi algoritam pretrage u KLEE-u

Projekat u okviru kursa
Verifikacija softvera
Matematički fakultet

Marija Mijailović 1093/2017
Miloš Lončarević 1034/2017
Filip Miljaković 1040/2017
Jelena Ivković 8/2013

19. Septembar 2018.

- **Naš zadatak bio je da ispitamo implementaciju alata KLEE, a potom i da implementiramo novi algoritam pretrage.**

- **Šta je KLEE?**
- **Šta je simboličko izvršavanje?**
- **Arhitektura**
- **Prolazak kroz stablo stanja**
- **Opis rešenja problema**

Šta je KLEE?

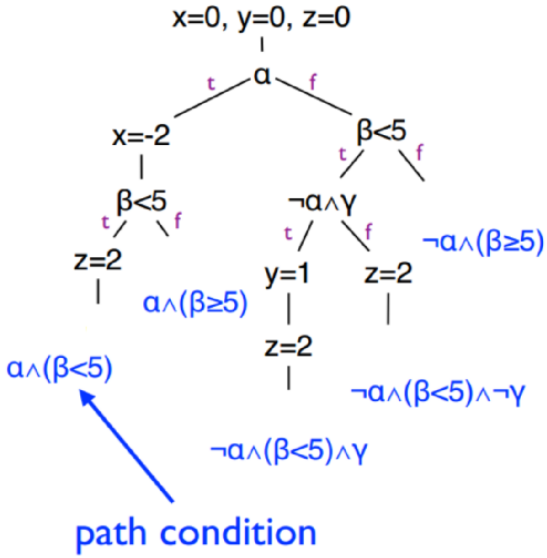
- KLEE je javno dostupan alat koji služi za simboličko izvršavanje programa i za automatsko generisanje test primera.
- Vrši analizu LLVM koda i koristi SMT rešavač STP za proveravanje uslova ispravnosti koje generiše.
- Ciljevi su pokrivenost svih linija izvornog koda programa i detekcija svih opasnih operacija, ako postoji ijedna ulazna vrednost koja može da prouzrokuje grešku.
- Simboličko izvršavanje automatski generiše ulaze za testiranje koji su proizvoljno izabrani simboli koji normalno učestvuju u svim izračunavanjima u okviru koda.

Šta je simboličko izvršavanje?

• Kada dođe do uslovnog grananja prilikom izvršavanja programa, u čijem uslovu učestvuju pomenuti simboli, sistem prati obe grane i generiše skup ograničenja (tzv. uslov putanje) koja moraju da važe u toj grani.

•U trenutku kada se u jednoj od grana otkrije greška, izabrani simbol dobija konkretan skup vrednosti na osnovu uslova te putanje u kojoj se greška dogodila.

```
int a = α, b = β, c = γ;
                // symbolic
int x = 0, y = 0, z = 0;
if (a) {
  x = -2;
}
if (b < 5) {
  if (!a && c)  { y = 1; }
  z = 2;
}
assert(x+y+z!=3)
```



Path 1: $\alpha = 1, \beta = 1$
Path 2: $\alpha = 1, \beta = 6$
Path 3 ...

Arhitektura

- Svakom simboličkom procesu je pridružen registarski fajl, stek, hip, programski brojač i uslov putanje. Ovakvu reprezentaciju simboličkih procesa nazivamo stanje.
- Ova stanja, za razliku od stanja normalnih procesa, predstavljena su u vidu stabla izraza gde su listovi simboličke promenljive ili konstante, a čvorovi su operacije u LLVM jeziku asemblera.
- KLEE prolazi kroz veliki broj ovih stanja tako što se izvršava u jednoj petlji koja određuje resdosled odabira stanja čije se vrednosti uzimaju pri simboličkom izvršavanju koda.

- **Petlja se izvršava sve dok se ne obiđe kod sa dvim stanjima ili dok ne istekne maksimalno zadato vreme za izvršavanje.**
- **Uslovi grananja su bulovski izrazi koji mogu imati vrednost *true* ili *false*, od kojih zavisi dalji tok programa.**
- **Takođe, KLEE može proveriti da li su ovi uslovi uvek zadovoljeni ili uvek nezadovoljeni i na osnovu toga ispratiti samo jedan tok programa, od moguća dva.**
- **U suprotnom, obe grane se moraju ispratiti pri čemu se kopira stanje simboličkih procesa.**

Prolazak kroz stablo stanja

- **Stablo stanja može biti veoma složeno pa je potrebno pronaći optimalan način za prolazak kroz sve delove stabla, a KLEE za to koristi sledeće dve heuristike:**
  1. **Nasumično biranje puteva – Sledeće stanje za izvršenje se bira tako što se putuje kroz binarno stablo od korena i na svakom grananju se nasumično bira putanja, tako da svaki skup stanja ima istu šansu da bude izabran bez obzira na veličinu podstabla.**
  2. **Pretraga bazirana na pokrivenosti – Izračunava koja stanja imaju najveću šansu da prođu kroz novi kod u bližoj budućnosti i na osnovu toga dodeljuje određenu težinu svim stanjima.**
- **KLEE ove dve strategije koristi naizmenično čime se ublažavaju mane pojedinačnih strategija i podiže ukupnu efikasnost.**

Opis rešenja problema

- U okviru KLEE-a postoji veći broj različitih vrsta pretrage, a naš zadatak je bio da osmislimo i implementiramo jedan novi algoritam pretrage, kao i da napravimo eksperimente kojima ćemo uporediti postojeće algoritme pretrage sa našim.
- Naša ideja je bila da stanja čuvamo u okviru AVL stabla, čiji čvorovi bi pored ExecutionState-a imali i pokazivače na levo i desno podstablo, kao i visinu podstabla čiji je taj čvor koren (neophodno zbog balansiranja stabla).
- KLEE-u prosleđujemo ono stanje koje je trenutno koren u AVL stablu.
- Odlučili smo se za čuvanje stanja u AVL stablima, jer nam to omogućava brze operacije umetanja, brisanja i pretrage.
- Kao kriterijum za pravljenje i balansiranje smo uveli novi parametar *nasWieght*, koji se nasumično postavlja za sve čvorove stabla.

Literatura

- https://llvm.org/pubs/2008-12-OSDI-KLEE.pdf
- http://www.programskijezici.matf.bg.ac.rs/vs/predavanja/05_simbolicko_izvrsavanje/05_simbolicko_izvrsavanje-prvi_deo_slajdovi.pdf
- http://www.programskijezici.matf.bg.ac.rs/vs/vezbe/07/vs_vezbe_07.pdf
- http://www.programskijezici.matf.bg.ac.rs/vs/vezbe/08/vs_vezbe_08.pdf
- https://github.com/tum-i22/klee22
- https://github.com/JelenaI/VS---projekat

Hvala na pažnji