Abstraktsete domeenide omaduspõhine testimine

Bakalaureusetöö

Simmo Saan

Tartu Ülikool, arvutiteaduse instituut

Juuni, 2018

Staatiline programmianalüüs

- Staatilise analüüsiga uuritakse programme ilma neid käivitamata. See võimaldab otsida vigu või tõestada teatud vigade puudumist.
- Üks võimalus on abstraktse interpretatsiooniga ligikaudsete programmi seisundite määramine. Vastav teooria garanteerib analüüsi korrektsuse (ingl. *sound*).
- Analüsaatorites endis esineb vigu, mistõttu tehtavad analüüsid ja nende korrektsus on rikutud.

Eesmärk

Goblint analüsaatori

- Domeenide omaduspõhine testimine
- Vigade tuvastamine

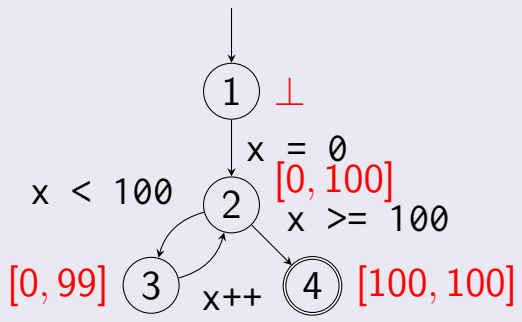
Intervallid

- Täisarvude staatiliseks analüüsiks saab kasutada **intervalle**
  - Näiteks $[0, 3]$, $[-1, 5]$, $[2, 2]$, $[1, +\infty]$, $[-\infty, +\infty]$
- Aritmeetilised tehted intervallidel
  - Näiteks liitmine $[0, 3] + [-1, 5] = [-1, 8]$
- Osalise järjestuse seos sisalduvuse kaudu
  - Näiteks $[2, 2] \sqsubseteq [0, 3] \sqsubseteq [-1, 5] \sqsubseteq [-\infty, +\infty]$
  - Kokkuleppeliselt väiksem tähendab täpsemat
- Ühendamise tehe ühendi kaudu
  - Näiteks $[0, 3] \sqcup [5, 7] = [0, 7]$
- Suurim intervall
  - $\top = [-\infty, +\infty]$

Näidisanalüüs intervallidega

Muutuja x väärtus

```
int x = 0;
while (x < 100)
    x++;
```



Täielikud võred

Domeen (võimalike väärtuste hulk) peab moodustama **täieliku võre**:

- Elementide hulk $\mathbb{D}$
- Osalise järjestuse seos $\sqsubseteq$
- Ülemise raja tehe $\sqcup$
- Alumise raja tehe $\sqcap$
- Suurim element $\top$
- Vähim element $\bot$

Omaduspõhine testimine

- Haskellis QuickCheck
- Hea matemaatiliste tingimuste kontrollimiseks
- Omadused — predikaadid
- Generaatorid — juhuslikud elemendid

Goblint analüsaator

- TÜ, TUM
- Mitmelõimelised C programmid
- Andmejooksud
- Kirjutatud OCaml-is

Goblinti täiendused

- Domeenidesse generaatorid
- Kõik omadused omaduspõhiste testidena
- Näiteks ülemraja kommutatiivsus ($a \sqcup b = b \sqcup a$):

```
make ~name:"join comm" (pair arb arb)
  (fun (a, b) ->
    D.equal (D.join a b) (D.join b a))
```

kus `D` testitav domeen, `arb` selle generaator

Ülevaatlikud tulemused

Lähenemine	Võrdlemine	Testide arv: Kokku	Erindi teke	Mittekehtivad
Alt üles	`equal`	468	~35	~75
	`leq`	468	~35	~69
Ülalt alla	`equal`	27	0	~12
	`leq`	27	0	~12

Erindite ja mittekehtimise põhjuseid

1. Viga domeeni implementeerimisel
2. Mittekehtimine teoreetilisel tasandil
3. Teadlik ja dokumenteeritud mittekehtimine
4. Sõltumine teistest probleemsetest omadustest/domeenidest

Trieri domeen

Trieri domeen on Goblintis vaikimisi kasutusel, kuid testimisel leiti selle disainist funamentaalne viga: ülemraja tehe pole assotsiatiivne.
Goblinti arendajad võtavad leitut väga tõsiselt.

Järeldus

Omaduspõhist testimist on võimalik efektiivselt rakendada abstraktsetest domeenidest vigade leidmiseks.