

Minimalna Lokalna Aloakcija Registra

Aleksandar Urošević

24. januar 2024.

Sažetak

Ovaj rad istražuje problem minimalne lokalne aloakcije registra, fokusirajući se na optimizaciju alokacije registra unutar blokova koda bez skokova. Cilj je pronaći alokaciju koja minimizuje troškove učitavanja i čuvanja podataka, koristeći ograničeni skup dostupnih registara. Ovaj problem je značajan za optimizaciju kompilatora i interpretera. Prethodni radovi na ovu temu pokazuju da je ovaj problem NP težak [1]. Postoje razna rešenja ovog problema, poput pretvaranja problema u rešavanje puzla [3]. U ovom radu se pravlja rešenje zasnovano na genetskom programiranju.

1 Opis problema

Problem o kojem se diskutuje je problem nalaženja najjeftinijeg načina izvršavanja bloka koda (bez skokova). Pretpostavljamo da imamo na raspolaganju N registara, i S_i ($1 \leq i \leq N$) troškove pri učitavanju ili čuvanju registra i . Za datu sekvencu instrukcija programa treba pronaći optimalnu strategiju učitavanja i čuvanja registra tokom izvršavanja programa.

Sledi opis problema jednim primerom (1). Ovaj program je identican kao primer iz literature [2].

Instrukcija programa *ADD* x, y, z (kao i ostale instrukcije) označava da je neophodno imati promenljive x, y učitanim u registre i neophodno je zapisati novi podatak u z . Ova instrukcija može da bude prevod linije koda $z = x + y$. Cena učitavanja vrednosti u registar je S_i . Ukoliko su svi registri puni, neophodno je sačuvati vrednost registra i zameniti je novom potrebnom vrednošću, cena ove procedure je $S_i + S_i = 2S_i$, jedno čitanje i jedno pisanje. U desnoj koloni zapisani su koraci koje bi procesor računara izvršio, predstavljen je optimalno rešenje dobijeno genetskim algoritmom. Ovaj rezultat je potvrđeno optimalan putem iscrpne pretrage.

Ilustracija

[illegible]

Slika 1: Primer 1

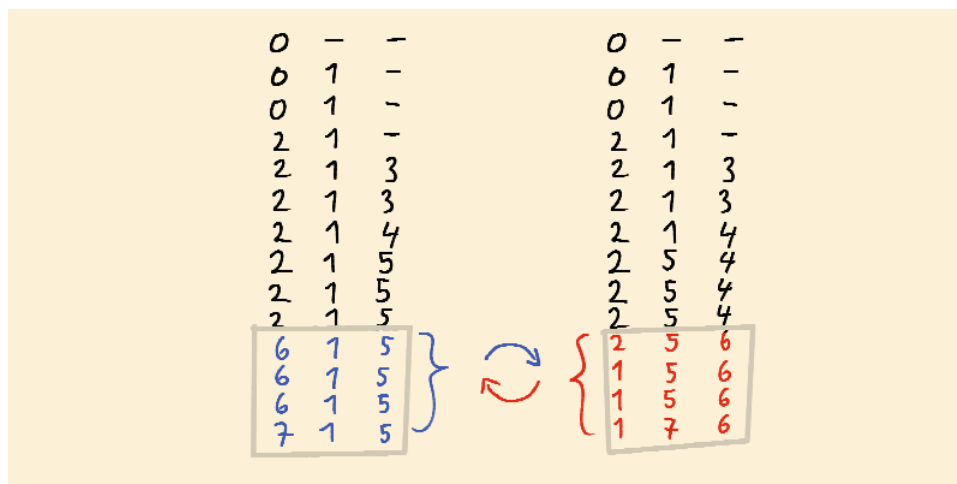
Postoje mnogo ranijih radova na ovom problemu. Od interesantnih radova je „Register Allocation by Puzzle Solving“ gde se diskutuje praktično rešavanje generalnog problema alokacija registra (ne samo lokalnog), i to prevođenjem problema na problem rešavanje puzzle[3]. Ovo daje ideju da redove instrukcija možemo da posmatramo kao blokove koje slažemo. Možemo osmisliti rešenje korišćenjem genetskog programiranja.

Funkcija cilja je suma cena korišćenja registara. Možemo je izračunati prola-zeći redom kroz niz alokacija registara rešenja, od prvog reda pa do posled-njeg, posmatrajući kada se i kako registri menjaju.

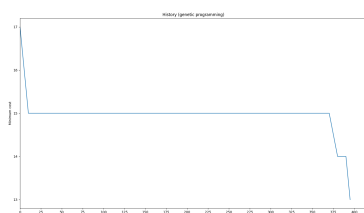
Ukrštanje implementiramo biranjem nasumičnog broja i td. $0 \leq i < N$, zamjenjujemo sve nizove alokacija A_j sa B_j za sve $i \leq j < N$ (2)

Mutacija je implementirana nasumičnim biranjem broja i td. $0 \leq i < N$ koji predstavlja polazni red programa od kojeg nasumično gradimo ostatak niza alokacije.

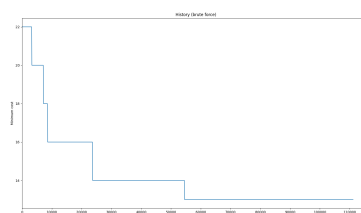
Ilustracija



Slika 2: Primer 2



(a) genetičko programiranje



(b) iscrpna pretraga

Slika 3: Primena genetičkog programiranja i iscrpne pretrage na primer 1

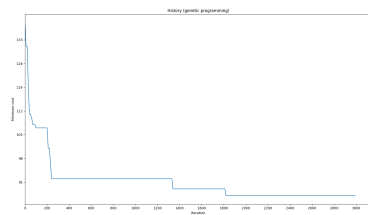
2 Rezultati

Primenom genetskog programiranja na prvi primer koji je dat i u literaturi [1] dobijamo optimalna rešenja za mali broj iteracija (uz sreću), iscrpa pretraga je takodje prikazana na figuri 3.

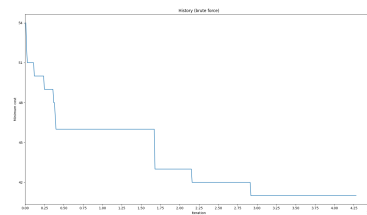
Na figuri 4 prikazani su rezultati za veći primer koji se ne nalazi u literaturi.

3 Zaključak

Korišćenje genetskog programiranja za nalaženje optimalnog načina alociranja registara možda ne daje rešenja brže od već razvijenih heurističkih algoritama [1] [3]. Razlog je nesigurnost u vremenu koji je potreban da se dostigne dobar rezultat. Ovo se može poboljšati drugim metodama selekcije,



(a) genetičko programiranje



(b) iscrpna pretraga

Slika 4: Primena genetičkog programiranja i iscrpne pretrage

mutacije i ukrštanja, ali nesigurnost i dalje ostaje. Razvijena metoda ovde diskutovana daje dobra rešenja, ali sporo u poređenju sa drugim metodama.

Literatura

- [1] M. Farach and V. Liberatore. On local register allocation. 1998.
- [2] Vincenzo Liberatore, Martin Farach-Colton, and Ulrich Kremer. Evaluation of algorithms for local register allocation. 1998.
- [3] Fernando Magno Quint~ao Pereira. and Jens Palsberg. Register allocation by puzzle solving. 1998.