SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

SEMINAR

Paralelni genetski algoritam - otočni model

Leon Novački
Voditelj: *Marko Đurasević*

Zagreb, svibanj 2022.

SADRŽAJ

1. Uvod

Genetski algoritam je prikladan algoritam pronalaska rješenja za mnoge probleme. No, povećanje potreba društva je dovela do potrebe poboljšanja algoritama koji traže rješenje, i time je stvorena i potreba za poboljšanje genetskog algoritma. Kompleksniji problemi imaju veći prostor rješenja koji je potrebno pretražiti, pa se time vrijeme izvođenja algoritma povećava. Jedan od najboljih načina smanjenja vremena trajanja izvedbe algoritma je njegova paralelizacija. Ovaj rad razmatra otočni model paralelizacije genetskog algoritma.

1.1. Genetski algoritam

Genetski algoritam je algoritam koji traži rješenje zadanog problema evoluirajući populaciju u kojoj je svaka jedinka jedno rješenje problema. Populacija sadrži fiksan broj jedinki od kojih stvaramo nove jedinke sljedeće generacije koristeći genetske operatore. Najčešće upotrebljavani genetski operatori su operatori seleckije, križanja i mutacije.

Algorithm 1 Genetski algoritam

```
while not uvjet_prekida do
    new_population = []
    while size(new_population) < pop_size do
        p1, p2 = selekcija(population)
        c = krizanje(p1, p2)
        c = mutacija(c)
        new_population ← c
    end while
    population = new_population
end while
```

Algoritam se izvršava dok se ne ispuni uvjet prekida algoritma. Uvjet prekida algo-

ritma može biti pronalazak prihvatljivog rješenja, istek vremena ili prijeđen dopušten broj iteracija genetskog algoritma.

2. Otočni model

Otočni ili distribuirani genetski algoritam je model paralelnog genetskog algoritma gdje je populacija rješenja podijeljena na subpopulacije ili otoke. Nad tim otocima genetski algoritmi paralelno razvijaju jedinke. Otočni model je najčešće korišteni model paralelnog genetskog algoritma jer ga je jednostavno implementirati, i ima uporište u prirodi gdje izolirane populacije iste vrste se paralelno razvijaju.

2.1. Osnovni otočni model

Osnovni otočni model, zvan i trivijalni paralelni genetski algoritam (Golub, 2004) je ustvari samo paralelno izvođenje samostalnih genetskih algoritama odjednom. Ako se traži jedno rješenje, kada jedan proces genetskog algoritma ostvari uvjet prekida izvođenja pošalje ostalim procesima signal za prekid jer je dovoljno dobro rješenje pronađeno.

Korist takvih osnovnih modela je u prikupljanju podataka za statističku analizu. Genetski algoritam je stohastički proces, pa je efikasan način prikupljanja podataka od velike važnosti. Statističkom analizom mogu se usporediti genetski algoritmi s različitim funkcijama selekcije, križanja, i mutacije, i mogu se usporediti genetski algoritmi s različitim numeričkim hiperparametrima kao veličina populacije, broj iteracija, vjerojatnost mutacije i drugi.

2.2. Operator migracije

Poboljšanje nad trivijalnim otočnim genetskim algoritmom se može ostvariti dodavanjem funkcije migracije. Funkcijom migracije najbolje jedinke jedne subpopulacije migriraju u druge subpopulacije.

Algorithm 2 Osnovni otočni paralelni genetski algoritam

```
while not uvjet_prekida do
  for num_process in parallel do
    new_population = []
    while size(new_population) < pop_size do
      p1, p2 = selekcija(population)
      c = krizanje(p1, p2)
      c = mutacija(c)
      new_population ← c
    end while
    population = new_population
  end for
end while
```

Algorithm 3 Otočni GA s migracijama

```
while not uvjet_prekida do
  for num_process in parallel do
    new_population = []
    if iteracija % interval_migracije == 0 then
      m = migracija()
      new_population ← m
    end if
    while size(new_population) < pop_size do
      p1, p2 = selekcija(population)
      c = krizanje(p1, p2)
      c = mutacija(c)
      new_population ← c
    end while
    population = new_population
  end for
end while
```

2.2.1. Migracijski interval

Funkcija migracije ne izvodi se svaku iteraciju genetskog algoritma nego tek svakih nekoliko iteracija. Interval iteracija između dvije migracije zove se *migracijski interval*. Može se koristiti i termin *frekvencija migracije* i ona je recipročna vrijednost migracijskog intervala.

Što je migracijski interval manji, to će se češće migracija izvoditi pa će time i subpopulacije genetskog algoritma biti međusobno sličnije.

Biranjem premalog migracijskog intervala dovodi do usporavanja izvedbe paralelnog genetskog intervala zbog potrebnog vremena za sinkronizaciju pojedinih genetskih algoritama i zbog potrebnog vremena za samu izvedbu razmjene jedinki pojedinih subpopulacija. Prečesta i spora komunikacija i sinkronizacija može dovesti do toga da paralelni genetski algoritam bude sporiji od ekvivalentnog jednoprocesnog genetskog algoritma (Budin et al., 1998).

2.2.2. Stopa migracije

Stopa migracije je hiperparametar paralelnog genetskog algoritma otočnog modela koji označava broj jedinki koji migriraju iz jedne subpopulacije u drugu subpopulaciju svaki migracijski interval. Velika stopa migracije će povećati sličnost između subpopulacija genetskog algoritma. Ako su subpopulacije slične, to znači da će više pojedinih paralelnih genetskih algoritama pretraživati istu okolinu prostora rješenja.

Najčešće je migracijska stopa jednaka jedinici, tojest, samo jedna jedinka migrira izvan i u svaku populaciju.

2.2.3. Migracijska selekcija

Migracijska selekcija je operator kojime biramo jedinku ili jedinke koje će migrirati iz subpopulacije. Kao i klasičan operator selekcije, migracijska selekcija vrši selekcijski pritisak i utječe na brzinu konvergencije.

Za operator migracijske selekcije može se koristiti isti operator kao i za klasičnu selekciju, ali odabir operatora migracijske selekcije, u kombinaciji sa migracijskim intervalom i stopom migracije može dovesti do *superlinearnog* ubrzanja.

Superlinearno ubrzanje

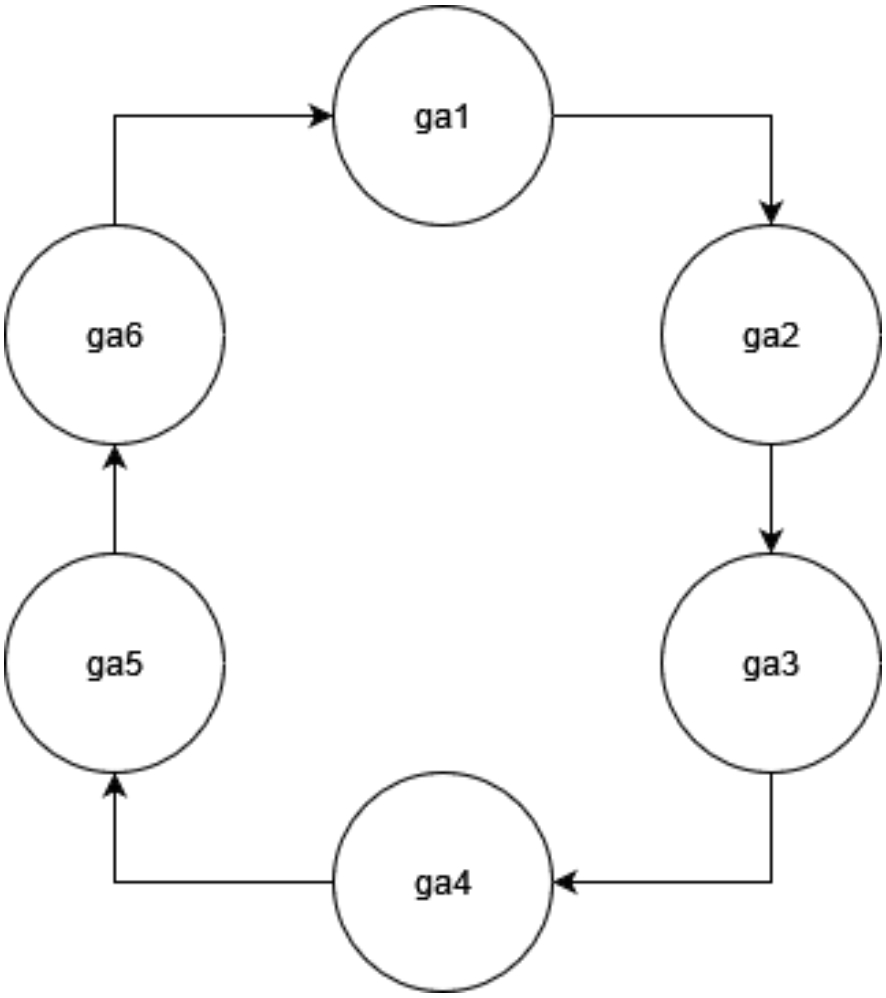
Superlinearno ubrzanje je ubrzanje dobiveno paralelizacijom koje je veće od broja paralelnih procesa, a uzrok je dodatan selekcijski pritisak koji migracije uzrokuju (Toma-

ssini, 1999). Istraživanja na promjenama genetskog sadržaja populacije su pokazala da se nova rješenja pronalaze ubrzo nakon izmjena jedinki između subpopulacija (Cantú-Paz, 2000).

2.3. Topologija migracija

Topologija migracija je dodatan hiperparametar otočnog modela paralelnog genetskog algoritma koji definira među kojim subpopulacijama se događa izmjena jedinki.

Najjednostavnija topologija za implemetirati je prstenasta topologija, gdje svaka populacija prima jedinke iz sebi prethodne populacije, a šalje jedinke sebi sljedećoj populaciji.



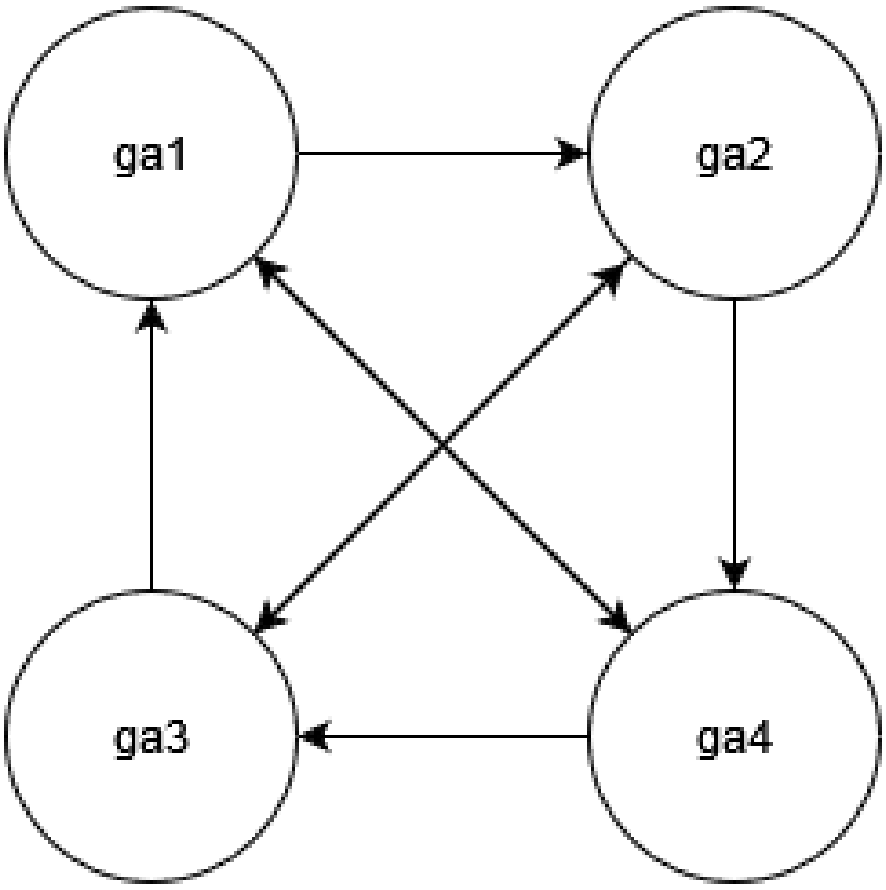
Slika 2.1: Prstenasta topologija

Jednostavne topologije kao prstenasta imaju manu toga da se dobro rješenje presporo širi po ostalim populacijama. Ako otok indeksa 1 napravi značajan napredak,

potrebno je 5 migracijskih intervala da taj napredak dođe do otoka indeksa 6. Zato prstenasta topologija loše iskorištava procesorsku moć.

Bolje povezanim topologijama se brže šire dobra rješenja, ali također je i vrijeme utrošeno na komunikaciju veće.

Migracije se unutar jednog migracijskog razdoblja ne moraju događati samo između dvije populacije, nego se mogu događati i među više njih.



Slika 2.2: Primjer bolje povezane topologije

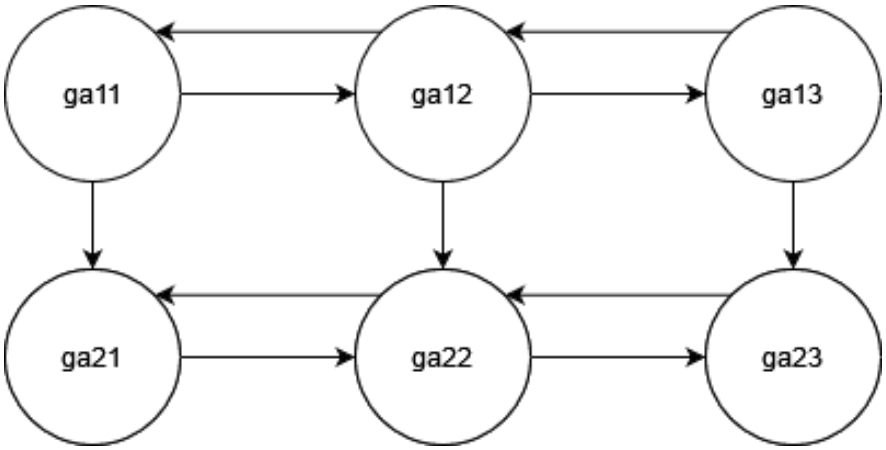
Prethodno navedene topologije su sve statičke. Osim statičkih, topologije mogu biti i dinamične.

Jednostavan primjer dinamične topologije je nasumična topologija. Prije migracije pojedini genetski algoritam nasumično odabire subpopulaciju kojoj će poslati jedinku.

Osim nasumične topologije, pod dinamičnu topologiju spadaju i topologije koje se tijekom izvođenja genetskog algoritma mijenjaju.

2.3.1. Višeslojna topologija

Višeslojne topologije koriste se jer smanjuju vjerojatnost da genetski algoritam zapne u lokalnom rješenju. Topologije viših slojeva migracijom šalju svoje najbolje jedinke u niže slojeve, ali se obrnuto ne događa. Homogenoj populaciji genetskog algoritma je često teško izaći iz svoje okolice. Višeslojna topologija osigurava priljev novih rješenja koja križanjem mogu izbaviti genetski algoritam iz lokalnog optimuma.



Slika 2.3: Primjer dvoslojne topologije

2.4. Prednosti i mane otočnog modela

Otočni model je najistraživaniji model paralelnog genetskog algoritma Cantú-Paz (2000). Jednostavno je postojeću implementaciju genetskog algoritma paralelizirati otočnim modelom i brzo postići traženo poboljšanje, pa je zbog toga i popularan.

Najveća mana otočnog modela je povećanje skupa hiperparametra koje dolazi s korištenjem migracije. Migracijski interval, stopa migracije, migracijska selekcija i topologija migracije sve utječu na jačinu selekcijskog pritiska i time brzinu konvergencije. Uz postojeće hiperparametre sekvecijskog genetskog algoritma, otočni model genetskog algoritma iziskuje još težu optimizaciju hiperparametara kako bi paralelizacija ostvarila poboljšanje.

Ovisno o vrsti problema, sitnozrnatija paralelizacija može biti prikladnija od otočnog modela. Prisutnost operacije migracije traži sinkroniziranost pojedinih genetskih algoritama kako bi se migracija odvila između dvije iteracije genetskih algoritama. Evaluacija jedinki je često vremenski najskuplja operacija u genetskom algoritmu, i ovisno o problemu, vrijeme evaluacije može biti varijabilno. U tom slučaju, genetski algoritmi koji se odvijaju brže od svojih susjeda će morati čekati da najsporiji genetski

algoritam dođe do iste iteracije kako bi se mogla odviti migracija. Takve praznine u radu pojedinih procesa se ne bi pojavile kod sitnozrnate paralelizacije genetskog algoritma.

3. Zaključak

Paralelizacija genetskog algoritma otočnim modelom je najjednostavnija implementacija paralelnog genetskog algoritma. Jednostavnost implementacije nažalost dolazi s velikim brojem hiperparametara koji se moraju optimizirati kako bi paralelni genetski algoritam dobro funkcionirao. Loše odabrani hiperparametri će dovesti ili do prerane konvergencije ili neće biti poboljšanja nad jednoprocesnim genetskim algoritmom. Također, efikasnost paralelizacije ovisi i o vrsti problema koja se rješava genetskim algoritmom, i u tim slučajevima drugačiji modeli paralelizacije genetskog algoritma su prikladniji.

4. Literatura

Leo Budin, Marin Golub, i Domagoj Jakobović. Parallel adaptive genetic algorithm. U *NC*, 1998.

Erick Cantú-Paz. A survey of parallel genetic algorithms. 2000.

Marin Golub. *Genetski algoritam - drugi dio*, 2004.

Marco Tomassini. Parallel and distributed evolutionary algorithms: A review, 1999.

5. Sažetak

Rad opisuje kako funkcionira distribuirani ili otočni model genetskog algoritma. Objašnjava hiperparametre migracijskog intervala, stop migracije, migracijsku selekciju i kako oni utječu na rad genetskog algoritma. Opisuje neke korištene topologije migracije i objašnjava kako različite topologije imaju utjecaj na svojstva genetskog algoritma.