Konkurentni i Distribuirani sistemi - Projekat

Distribuirani haos

– Dokumentacija –

Dimitrije Šejat

28/18-RN

Računarski fakultet 2022.

Sadržaj

[1. Uvod 2](#_gjdgxs)

[2. Konfiguracija čvora 3](#_30j0zll)

[3. Arhitektura sistema 3](#_1fob9te)

[3.1. Bootstrap server 3](#_3znysh7)

[3.2. Arhitektura 4](#_2et92p0)

[4. Algoritam 4](#_tyjcwt)

[4.1. Ulazak čvora u mrežu 4](#_3dy6vkm)

[4.2. Raspodela posla 4](#_1t3h5sf)

[4.3. Iscrtavanje 4](#_4d34og8)

[4.4. Otkaz čvora 5](#_2s8eyo1)

[5. Otpornost na otkaze 5](#_3rdcrjn)

[6. Poruke 6](#_26in1rg)

# Uvod

Ovim dokumentom predstavljena je tehnička dokumentacija distribuiranog sistema koji računa fraktalne strukture primenom Igre Haosa. Sistem korisniku omogućava sledeće:

* Pokretanje izračunavanja za jedan fraktal
* Dobijanje kompletne slike fraktala
* Otpornost na otkaze
* Sistem ne mora da podržava istovremeno izvršavanje operacija na različitim čvorovima. takođe, ne treba podržavati istovremeno uključivanje ili isključivanje čvorova.

# Konfiguracija čvora

Pri pokretanju čvora, automatski se iščitava konfiguraciona datoteka u kojoj se navode sledeći atributi:

* Port na kojem čvor sluša
* IP adresa i port bootstrap čvora
* Slaba granica otkaza
* Jaka granica otkaza
* Skup predefinisanih poslova. Svaki posao ima sledeće atribute:
  1. Naziv – String koji predstavlja jedinstveno ime za posao
  2. N – broj tačaka fraktalne strukture, Nϵ(3,10)
  3. P – udaljenost između trenutne tačke i odredišta gde će se pojaviti nova tačka
  4. W, H – dimenzija površine na kojoj se računaju tačke predstavljena kao int
  5. A – skup N tačaka tj niz od N parova int-ova

# Arhitektura sistema

## Bootstrap server

U sistemu postoji bootstrap server, koji nije čvor u mreži. On se koristi da novom čvoru, prilikom uključivanja u mrežu, prosledi podatke o nekom čvoru koji se već nalazi u mreži. Čvor prilikom uključivanje šalje hail poruku bootstrap serveru, na šta mu on šalje adresu i port nekog nasumičnog čvora iz sistema, kako bi mu se javio radi dalje organizacije sistema. Čim bootstrap prosledi potrebne podatke novom čvoru, komunikacija između njih se prekida i bootstrap nije svestan dalje organizacije sistema.

* Hail - tekst ove poruke nije bitan
* Contact - tekst ove poruke je kontakt koji se prosleđuje ili ništa ukoliko još nema čvorova u sistemu

## Arhitektura

Pri implementaciji sistema korišćen je kompletan graf, odnosno svaki čvor je povezan sa svakim drugim čvorom u mreži. Dakle, komunikacija između čvorova je uvek direktna.

# Algoritam

## Ulazak čvora u mrežu

Ukoliko je čvor koji se priključuje mreži prvi u sistemu, ništa se ne dešava, a svaki sledeći čvor šalje NewNodeMessage nekom od čvorova koji je već u sistemu, a čije podatke je dobio od bootstrap servera. Čvor koji je dobio NewNodeMessage treba da proveri koliko njih je trenutno u sistemu, da bi na osnovu toga organizovao sistem.

## Raspodela posla

Na početku u servent.properties postoji posao koji treba da se crta i zapamćen je u AppConfig-u (širina, visina, broj tačaka). Kada dođe novi čvor, on gleda ako je broj čvorova koji je trenutno u sistemu jednak broju tačaka koje ta slika ima, onda će se taj posao podeliti na čvorove.

Posao se deli uvek na isti način. Prolazi se kroz sve tačke i za svaku tačku uzimamo nju samu i tačke koje se nalaze na proporciji rastojanja od te tačke do ostalih tačaka. Isto važi ako je trougao, za 5,7,9 itd kada je deljivo sa broj čvorova – 1, izuzev prva 3 čvora.

Takođe postoji metoda nextColleagueToSplit kako bi se znalo koji deo sledeći da se podeli, da bi se delilo ravnomerno.

## Iscrtavanje

Čvor koji je dobio obaveštenje da je stigao novi čvor, organizuje ko će crtati koji deo slike, tj koji fraktal. On napravi mapu gde je ključ port čvora, a vrednost lista tačaka koja predstavlja njegov frame tj okvir unutar koga on crta. Zatim to šalje svim ostalim čvorovima da bi svi znali ko šta crta.

Postoji nit PictureDrawWorker koja ima komandu startDrawCommand koja svima kaže da započnu sa crtanjem. Svaki čvor tada u niti stavi boolean na true i započne se crtanje.

Prilikom organizovanja sistema svako ima svoj Frame i zna unutar kojih granica treba da crta. U AppConfigu postoji niz tačaka, koji ukoliko je prazan, označava da se prvi put započinje crtanje. Tada se unutar frame-a bira jedna nasumična tačka i od nje se počinje.

Osim komande start, postoje i komande stop i resault. Komanda stop govori svima da prestanu sa crtanjem, a komanda resault šalje svima requestResault. Kao odgovor na requestResault, čvorovi šalju sendResault, čime čvoru koji traži rezultate šalju svoje tačke koje su iscrtali. Ovaj čvor zatim sve prikupljene rezultate ubaci u niz allDots u AppConfig-u i iscrta ga pomoću BufferedImage-a.

## Otkaz čvora

Pored PictureDrawWorkera postoji i PingPongWorker nit, koja se pokreće čim se startuje neki čvor. Ona na svake 4 sekunde šalje pingove svim čvorovima. Na svaki peti ping, čvor svim svojim kolegama pošalje svoj backup, tj svoj niz tačaka, da u slučaju otkaza podaci ne budu izgubljeni. U AppConfig-u postoje backup-i za sve. To je zapravo mapa gde je ključ Integer što predstavlja port čvora, a vrednost je lista tačaka, što su zapravo svi ti podaci od nekog čvora.

Kada pošaljemo ping i od nekog čvora nam stigne pong, njega ubacimo u niz pongResponses u AppConfigu (predstavlja niz za odgovore). Ukoliko nakon isteka 4 sekunde postoji port koji se nalazi u svim kolegama, a nema ga u pongResponses, to znači da on nije poslao pong, tj nije odgovorio. Tada se on stavlja u niz suspiciousPorts i radi se provera sa ostalim čvorovima da li je on zaista otkazao. Provera se radi tako što nadjemo prvi čvor među našim kolegama koji nije u suspiciousPorts i njemu pošaljemo SuspiciousPortMessage i od njega tražimo proveru. Taj čvor tada pošalje suspiciosPing tom čvoru koji deluje da je otkazao. Po istom principu kao i ranije, čeka se odgovor 4 sekunde i ako se ne dobije, onda je taj čvor i dalje suspicious i inicijalnom čvoru se javlja da nije dobijen odgovor. U tom trenutku čeka se da istekne i jaka granica otkaza (10 sekundi) i ako za to vreme i dalje nije stigao pong, smatra se da je čvor otkazao i ubacuje se u niz brokenNodes u AppConfig-u.

# Otpornost na otkaze

Nakon otkaza neophodna je reorganizacija sistema. Reorganizaciju vrši čvor koji je primetio otkaz. On tada svim čvorovima šalje poruku sa obaveštenjem da je čvor sa odgovarajućim portom otkazao, kako bi ga svi ostali čvorovi ubacili u brokenNodes niz. Nakon toga, on svima šalje StopDrawingMessage, kako bi na kratko prestali da crtaju, dok se sistem reorganizuje, zatim restartuje svoje kolege i frame unutar koga crta i po istom principu kao na početku deli posao. Formira se nova mapa čiji je ključ port a vrednost lista tačaka koje predstavljaju frame za crtanje. Po završetku se svima šalje poruka NewSetup koja sadrži mapu sa novom raspodelom posla i StartDrawing, kako bi se ponovo započelo crtanje.

# Poruke

Svaka poruka ima senderPort i receiverPort, koji predstavljaju portove pošiljaoca i primaoca poruke

1. Backup Message (int senderPort, int receiverPost, List<Dot> dots)
   1. dots - niz tačaka koje neki čvor šalje svom kolegi kao backup (sve tačke koje je do sada nacrtao)
2. NewNodeMessage (int originalSenderInfo, int receiverInfo, String messageText, Map<Integer, List<Dot>> pictureParts)
   1. messageText - port čvora koji se dodaje u sistem
   2. pictureParts - map koja određuje koji kolega je zadužen za koji frame
3. NewSetupMessage (int senderPort, int receiverPort,Map<Integer, List<Dot>> colleagues)
   1. colleagues - nova mapa koja sadrži koji će kolega u kom frame-u da crta
4. NodeBrokenMessage (int senderPort, int receiverPort, String port)
   1. port - port čvora koji je pukao
5. SendResultMessage(int senderPort, int receiverPort, List<Dot> currentDots)
   1. currentDots - niz tačaka koje je dati čvor nacrtao
6. SuspiciousPortCheckedMessage(int senderPort, int receiverPort, String port, boolean stillSuspicious)
   1. port - port čvora koji je sumnjiv
   2. stillSuspicious - da li je čvor i dalje sumnjiv ili nije
7. SuspiciousPortMessage(int senderPort, int receiverPort, String port)
   1. port - port čvora koji je sumnjiv
8. WelcomeMessage(int senderPort, int receiverPort, String ipAddress, Map<Integer, List<Dot>> pictureParts)
   1. ipAddress - ip adresa koju novi čvor treba da doda kod sebe
   2. pictureParts - mapa koja predstavlja koji kolega će u kom frame-u crtati