

ChaosP2P Dokumentacija

Pavle Vodopija RN101/2018

Uvod	2
Proces rada	2
Bootstrap	3
Poruke	3
BasicMessage	3
JoinMessage	3
HelloMessage	3
WelcomeMessage	4
WorkMessage	4
DhtGetMessage	4
DhtGetResultMessage	4
DhtDisconnect	4
Protokol	5
Priključivanje	5
Podela posla	5
Distribuirana Hash Tabela	5
Primer rutiranja (DHT)	6

Uvod

Ovaj dokument predstavlja tehničku dokumentaciju za distribuirani sistem koji vrši simulaciju igre haosa. U stvari sistem je napravljen tako da uz ne puno promena može da zameni simulaciju nekim drugim sličnim poslom. Međutim, rezultat ovog sistem u trenutnom stanju je jedan ili više fraktala generisanih po pravilima igre haosa.

Proces rada

Svaki čvor pri pokretanju programa ne radi ništa i čeka **join** komandu od korisnika da se priključi nekom poslu. Nakon priključivanja proizvoljnim poslovima korisnik može da zatraži **status** ili **rezultat** odgovarajućom komandom. U slučaju da korisnik više ne želi da mašina radi na nekom poslu može prestati sa komandom **disconnect**. I na kraju, ako korisnik želi u potpunosti da zaustavi program koristi komandu **stop**. Detaljnije o komandama:

- **join [jobID]** - čvor se pridružuje mreži koja distribuirano izvršava posao sa jedinstvenim imenom **jobID**. Takođe, čvor može da bude u više različitih mreža (poslova) istovremeno.
- **disconnect [jobID]** - čvor se isključuje sa mreže **jobID**.
- **status [jobID] [fractalID]?** - ispis stusa za posao **jobID** i opcioni čvor na toj mreži sa identifikatorom **fractalID**.
- **result [jobID] [fractalID]?** - izvršava *snapshot* posla **jobID**, ili samo čvora **fractalID** ako je naveden, i exportuje rezultat kao sliku.
- **pause [ms]** - privremeno zaustavlja čitanje sa komandne linije.
- **stop** - zaustavlja rad programa.

Bootstrap

Bootstrap je specijalan program kome se novi čvorovi javljaju samo u slučaju priključivanja ili isključivanja sa mreže. On sam po sebi nije svestan veza u grafu neke mreže, već samo postojećih čvorova.

Komunikacija sa bootstrap-om je moguća samo preko tri tipa poruka:

JOIN i **DISCONNECT**. O ovome će biti rečeno više u odeljku [Protokol](#).

Bootstrap prati formulu za biranje čvorova koja zavisi od broja uglova generišućeg oblika **K**. Na primer za ($k = 3$), tj. trougao formula generiše sledeći niz:

0, 1, 2, |0, 0, 1, 1, 2, 2, |0, 0, 1, 1, 2, 2, 3, 3, 4, 4, 5, 5, 6, 6, 7, 7, 8, 8|, 0, 0, 1, 1, ..., $k^3 - 1, k^3 - 1$

Može se uočiti da je korak uvek $K - 1$ a ciklus se ponavlja na $k^i - 1$.

Poruke

U ovom odeljku su opisani svi različiti tipovi poruka koje su neophodne da bi sistem radio po protokolu.

BasicMessage

Sve poruke nasleđuju ovu poruku. Ona sadrži najosnovnije informacije kao što su: **fractalID**, **ip adresa** i **port** kako pošaljioaca tako i primaoca, **jobID** posla kome poruka pripada, **messageID** i **text** poruke.

JoinMessage

Koristi se pri priključivanju na neku mrežu i ima dva parametra: **fractalID** ako smo neki od prvih **K** čvorova i **sayHelloList** listu čvorova kojima treba da se javi kada se priključuje na mrežu.

HelloMessage

Ovo je prva poruka koja se šalje nekom čvoru kako bi se uvezali i razmenili najosnovnije informacije.

WelcomeMessage

Dobijamo je nakon što smo poslali [HelloMessage](#) i ona u sebi sadrži **fractalID** koji nam je dodelio čvor kome smo se javili da bismo ušli u mrežu.

WorkMessage

Šalje se nakon podele posla među novim čvorovima i služi kao signal da počinjemo sa radom. U sebi nosi podatke o dodeljenom poslu. U slučaju igre haosa, poruka sadrži glavne tačke **mainPoints** nad kojima se izvršava algoritam.

DhtGetMessage

Služi da bi podržala mehanizam iza funkcije **get()** u **Distribuiranoj Hash Tabeli** (više informacija u tom odeljku). U sebi sadrži **key** za željenu vrednost u tabeli.

DhtGetResultMessage

Poruka koja propagira kroz mrežu dok ne dodje do prvobitnog okidača **get()** funkcije i u sebi sadrži njen rezultat.

DhtDisconnectMessage

Po prijemu ove poruke raskida se veza sa pošaljiocem.

ReplaceNodeMessage

Ova poruka se šalje Bootstrap serveru kada čvor želi da napusti odredjeni posao i u njoj se nalaze sve neophodne informacije da novoodabrani čvor preuzme njegovo mesto.

PingMessage

Poruka se periodično šalje susedima i služi za detekciju otkaza. Ako čvor ne odgovori u određenom vremenskom intervalu, započinje se rebalansiranje grafa.

Protokol

Priključivanje

Prvo se šalje [JoinMessage](#) Bootstrap serveru i onda postoje dva slučaja:

1. Ako smo neki od prvih K čvorova dobićemo listu prethodnih K čvorova kojima treba da se javimo kako bismo formirali kompletan graf.
2. U suprotnom dobijamo tačno jedan čvor kojem ćemo se javiti. On će postati naš roditelj.

Nakon odgovora, javljamo se odgovarajućim čvorovima sa [HelloMessage](#) i tu opet postoje dva slučaja:

1. Ako nemamo **fractalID** dobijamo poruku [JoinMessage](#) od našeg roditelja sa dodeljenim **fractalID**-jem i listom komšija kojima se javljamo.
2. U suprotnom dobijamo [WelcomeMessage](#) od naših komšija.

Podela posla

U igri haosa, podela posla se vrši geometrijskim deljenjem na K jednakih celina. Kada roditelj neke pod mreže zalkjuči da se posao može podeliti, tj. kada je broj dece ili komšija jednak $(K - 1)$, on njima šalje [WorkMessage](#) sa svim potrebnim podacima da započnu izvršavanje svog dela.

Distribuirana Hash Tabela

Svaki čvor ima podršku distribuirane tabele vezane za specifičan posao. Bilo koju stavku iz tabele dobijamo pozivom **get(key)** funkcije sa bilo kojeg čvora za bilo koji čvor sa **fractalID**-jem kao **key**, a sve ostalo rešava uz mehanizam rutiranja i niz poruka [DhtGetMessage](#) i [DhtGetResultMessage](#).

Primer rutiranja (DHT)

Neka su $V(G) = \{ 00, 10, 20, 01, 02 \}$ čvorovi u neusmerenom grafu G i neka su čvorovi 01 i 02 deca čvora 00 i neka su 00, 10 i 20 komšije.

Ako bi čvor 10 pozvao $\text{get}(02)$ algoritam bi primenio sledeće korake:

1. $10 \rightarrow X \rightarrow 02$ - **mismatch** na prvom karakteru, šaljemo [DhtGetMessage](#) roditelju ako postoji, u suprotnom komšiji kome **fractalID** počinje sa 0 jer smo odsekli dvojku.
2. $10 \rightarrow 00 \rightarrow Y \rightarrow 02$ - **mismatch** na drugom karakteru, u proveru dece šaljemo [DhtGetMessage](#) 02.
3. $10 \rightarrow 00 \rightarrow 02$ - Sada je **fractalID** ovog čvora jednak ključu **key** što znači da šaljemo [DhtGetResultMessage](#) koja propagira istom putanjom do početnog čvora 10.

