

Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

Uporaba metode Monte Carlo za izračun Pi-ja v okolju Hadoop

Avtor: Matic Teršek
Mentor: Matevž Jekovec
datum: 28.9.2014

Uvod

V tej nalogi je bilo potrebno uporabiti metodo Monte Carlo, za aproksimacijo vrednosti Pi. Metoda Monte Carlo je metoda, ki se zanaša na ponavljajočo naključno vzorčenje podatkov za pridobivanje rešitev. Tako pridemo pri naši nalogi do težave velike količine podatkov. Zato smo uporabili okolje, ki je bilo razvito posebej za takšne naloge. Imenuje se Hadoop, katerega distributer je Apache. Hadoop je okolje, ki vhodne podatke razbije na več manjših kosov, kateri so nato vsaki posebej posredovani manjšim procesorjem. Tako imamo mrežo manjših procesorjev, ki vsak prejme manjšo količino podatkov in jih obdela. Tako pridobimo na času delovanja za vsak nov procesor ki ga dodamo. V Hadoop-u lahko procesorje združimo za reševanje enega problema s pomočjo programov MapReduce. Ti nam najprej vhodne podatke razbijejo, mapirajo, zložijo in nato zmanjšajo na količino izhodnih podatkov.

Delovanje moje implementacije

Ker v tem problemu vhodnih podatkov ni, je bilo potrebno podatke ustvariti. Tako sem prebral o Haltonovi sekvenci, kismet jo uporabil za generiranje točk v kvadratu.

Nato sem v map-u vsako točko preveril, ali pripada vrisanemu krogu in če pripada, povečamo števec pozitivnih v nasprotnem primeru pa števec negativnih. Potem iz vsakega Mapperja ta dva podatka prenesemo v Reducer, kjer jih reduce sešteje.

Nato v glavni metodi podatke še samo preberemo, ter na 20 decimalnih mest natančno izpišemo ocenjen Pi po formuli $Pi = 4 * (\text{pozitivni} / (\text{pozitivni} + \text{negativni}))$.

Meritve

Opravil sem več meritev. Najprej sem opravljal meritve z zaokroženim številom vzorcev (zaokroženo na 10 tisoč), zaradi česar so bili rezultati nekoliko čudni, kar je bilo po premisleku popolnoma razumljivo. Namreč ni pomembno katero vrednost delimo s 160 tisoč, slej kot prej se bo deljenje izteklo. Tako sem dobil naslednjo rešitev:

Število map = 400

Primerkov v posamezni mapi = 400

Ocenjena vrednost Pi-ja je: 3.14127500000000000000

Program zaključen v 1675.654 sekundah.

Zato sem v naslednji iteraciji spremenil število iteracij in dobil naslednje rešitve:

Število map = 1

Primerkov v posamezni mapi = 1985463100

Program zaključen v 92.767 sekundah.

Ocenjena vrednost Pi-ja je: 3.14159257857776354544

Število map = 100

Primerkov v posamezni mapi = 19854631

Program zaključen v 454.604 sekundah.

Ocenjena vrednost Pi-ja je: 3.14159254432882686160

Ker imam Hadoop postavljen kot enojni node, je časovna razlika popolnoma razumljiva saj je 100 krat pognati metodo map na enem vozlišču časovno zahtevnejše kot pa pognati en program ki bo ustvaril isto količino podatkov. Drugačna bi bila zgodba, če bi postavili hadoopov strežnik kot gručo vozlišč.

Največji izzivi

Ker sem prej že izdeloval nalogo o iskanju kvadratnega korena, sem celotno nalogo že razumel, tako da sem moral izdelati samo dve stvari. Generator naključnih točk in vstaviti formulo za določanje vsake točke. Pravilo za določanje točke sta mi profesor in asistent dobesedno povedala, tako da s tem delom naloge ni bilo zapletov, za generator pa sem na srečo naletel na Haltonovo sekvenco, ki je to težavo odlično rešila.

Zaključek

Mislilim da je bila to naloga, ki je v času mojega študija potrebovala največ moje pozornosti. Tako da sem na koncu same naloge zelo zadovoljen. V CV bom lahko dodal novo znanje in sicer osnovno razumevanje hadoopa, prav tako pa je bilo odlično spet razmigati moje možgane zaradi iskanja odločitvenega pogoja.