Eksperimenti jakog I slabog skaliranja   
Kanonov algoritam množenja matrica

### Tehničke karakteristike sistema

Operativni sistem: Windows 10

Dodatne biblioteke: numpy 1.21.2, mpi4py 3.1.1

# Analiza koda

Postupak: Procenat koda koji se može paralelizovati (p) dobijen je tako što se vreme izvršavanja ovog dela koda podeli ukupnim vremenom izvršavanja sekvencijalne verzije. Procenat koda koji ne može biti paralelizovan je onda 1 – p.

Merenjem vremena izvršavanja sekvencijale verzije za ulazne matrice dimenzija 200x200 dobijen je procenat sekvencijalnog dela koda koji se može paralelizovati i iznosi 99%, dok preostalih 1% predstavlja procenat koda koji ne može biti paralelizovan.

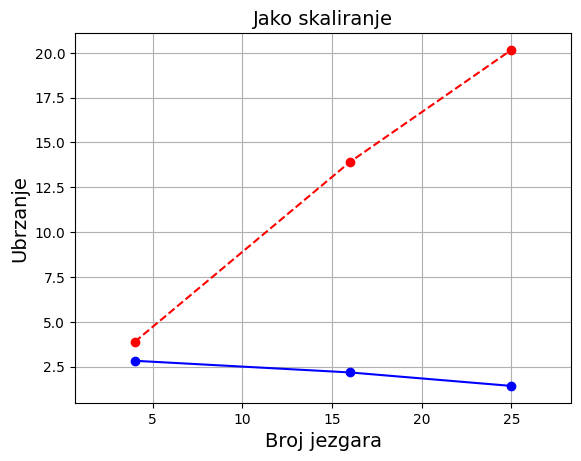
## Jako skaliranje, Amdalov zakon

Po Amdalovom zakonu maksimalno ubrzanje uslovljeno je delom koja koji se ne može paralizovati.

Jednačina maksimalnog ubrzanja: , gde s predstavlja procenat vremena ižvršavanja nužno sekvencijalnog dela koda, p procenat vremena izvršavanja koda koji može biti paralelizovan, n broj procesorskih jezgara.

Primenom Amdalovog zakona za problem Kanonovog algoritma na mašini sa 4 procesorska jezgra, dobije se maksimalno ubrzanje od 3.88 puta.

Eksperiment jakog skaliranja izveden je pokretanjem paralelne verzije koda sa ulaznim matricama dimenzija 200x200 na različitom broju procesorskih jezgara, 30 puta radi bolje preciznosti. Kako je ideja Kanonovog algoritma da u svakom koraku množi i siftuje podblokove dimenzija n/sqrt(p), paralelna verzija algoritma radi sa najmanje 4 procesa, jer je to najmanji broj p koji može da se korenuje. Za potrebe iscrtavanja grafika i samog eksperimenta, razmotren je rad algoritma i sa 16 i 25 procesorskih jezgara iako arhitektura sistema ne podržava taj broj procesora. Dobijeni rezultati predstavljeni su na grafiku ispod, x osa označava broj procesora, y osa postignuto ubrzanje. Ubrzanje je izračunato po formuli ts/tp gde je ts srednje vreme ižvršavanja sekvencijalne verzije, dok je tp srednje vreme izvršavanja paralelne verzije. Rezultati svakog od 30 pokretanja nalaze se u fajlovima “paralell4.txt”, “parallel16.txt” I “paralell25.txt”. Na grafiku crvenom bojom je označeno maksimalno ubrzanje dobijeno Amdalovim zakonom, zbog korišćenja 16 i 25 procesora, izračunato je maksimalno ubrzanje i za taj broj procesora, mada jasno je da je to ubrzanje na mašini od 4 procesora nemoguće. Maksimalno ubzanje dobijeno u ovom projektu postignuto je za 4 procesora i iznosi 2,83. Zbog kreiranja i pokretanja većeg broja procesa kada to tehnički nije moguće (16 i 25 procesora), ubrzanje je manje nego sa 4 procesora. Za 16 procesora ubrzanje je 2.18, dok za 25 iznosi 1.43. Odnos maksimalnog ubrzanja i ubrzanja postignutog u ovom projektu najbolje vidimo na prvoj tački grafa, koja predstavlja ubrzanje za 4 procesorska jezgra.



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Broj procesora | Serijalna sv | Paralelna sv | Serijalna std | Paralelna std | Ubrzanje |
| 4 | 39.82 s | 14.08 s | 5.39 | 3.27 | 2.83 |
| 16 | 39.82 s | 18.28 s | 5.39 | 4.39 | 2.18 |
| 25 | 39.82 s | 27.91 s | 5.39 | 11.96 | 1.43 |

U tabeli su za svaki od broja procesora, vidimo srednja vrednost izvršavanja sekvencijalne i paralelne verzije (oznaka sv), vrednosti standardne devijacije kod sekvencijalne i paralelne verzije kao i postignuto ubrzanje.