

Redas Jatkauskas
Informatikos katedra
III kursas

Lygiagretūs skaičiavimai
N-kūnų simuliacijos atvejis

Problemos formulavimas

*14. N-kūnų simuliacijos atvejis.

Kūnai traukia vienas kita pagal Niutono dėsnį. Jėga yra proporcinga:

- Kai $l > r_1$ ir $l > l_2$: $m_1 * m_2 / l^2$
- Kai $l < r_1$ arba $l < r_2$: $m_1 * m_2 / l$

Kur l – atstumas tarp kūnų, r – kūnų dydžio spindulys, m – kūnų masė

Lygiagretusis algoritmas

Programa sukasi ciklu nurodytą kiekį iteracijų, kur kiekvienoje vyksta du etapai:

1. Apskaičiuojama visų kūnų veikiančios jėgos vektorius,
2. Atnaujinami kūnų greičiai bei pozicijos pagal juos veikiančias jėgas.

Kiekvienas etapas vykdomas lygiagrečiai.

Po kiekvieno etapo yra barjeras, kuris užtikrina, kad pozicijos bus atnaujinamos tik tada, kai bus apskaičiuoti visos objektus veikiančios jėgos ir kitos iteracijos jėgos bus pradėtos skaičiuoti tik, kai visų objektų pozicijos bus atnaujintos.

Skaičiavimas vykdomas kiekvienai gijai ciklu kreipiantis į **Gravity** klasės objekto metodą **nextJob** iki tol, kol jis grąžins neigiamą binarinę reikšmę (**false**) – ženklą, jog darbas baigtas. Šis metodas viduje nusprendžia, ką skaičiuoti. Vidinis darbas yra paslėptas.

```
public void run() {  
    while (gravity.nextJob(threadID));  
}
```

nextJob viduje vyksta toks darbas:

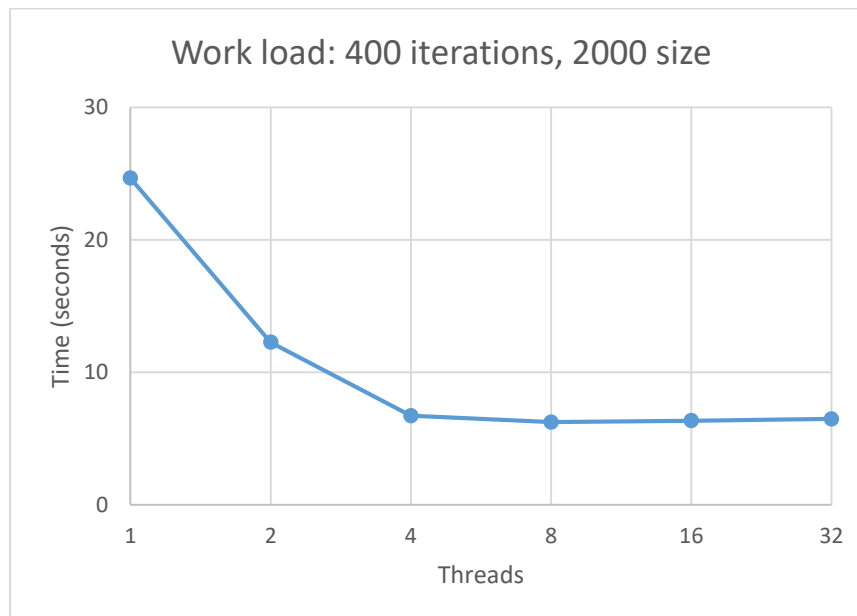
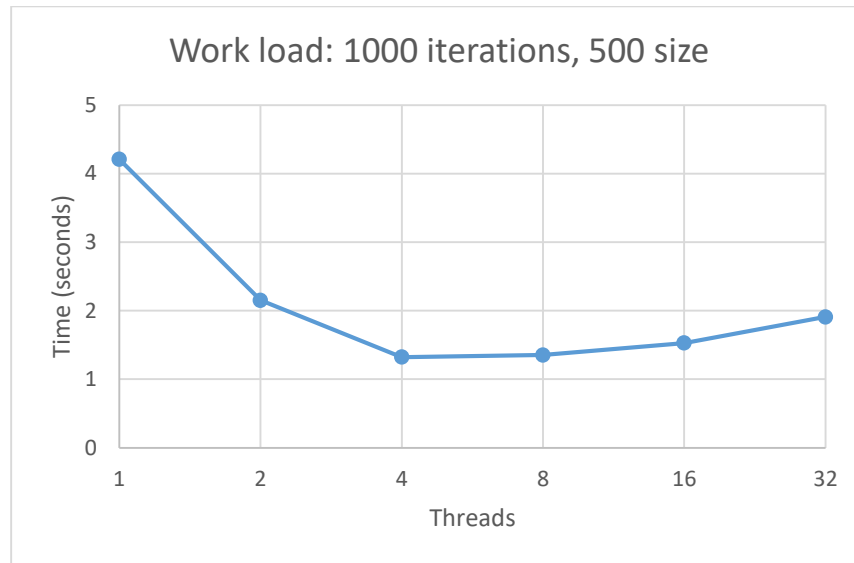
- Sinchroniškai gaunamas indeksas sekančio objekto, kuriam reikia atlikti skaičiavimus. Jeigu esamam etapui nėra laisvų objektų, metodas baigia darbą.
- Patikrinama, ar neįvykdytos visos iteracijos. Jei taip, grąžinama **false**.
- Atliekamas skaičiavimas.
- Pažymima, kad darbas atliktas, grąžinama **true**.

Vykdomo aplinka

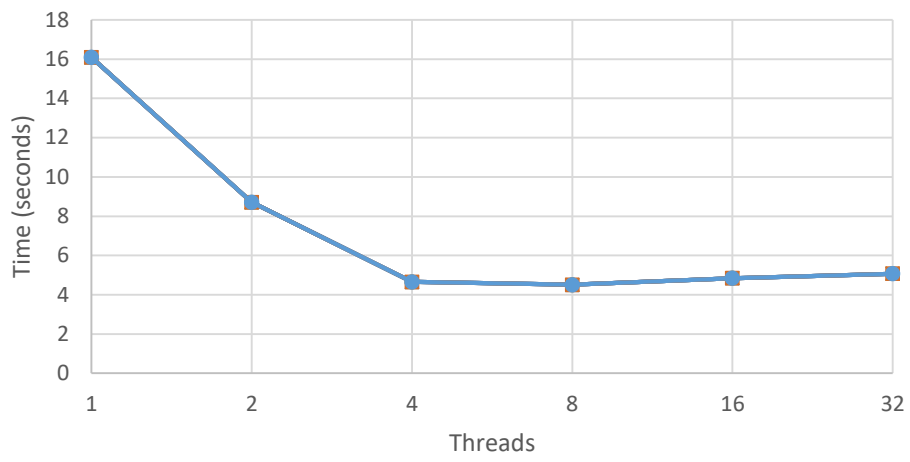
Procesorius: Intel(R) Core(TM) i7-4710HQ CPU @ 2.50GHz (**8 CPUs**), ~2.5GHz

Naudojama Java SDK 11 versija

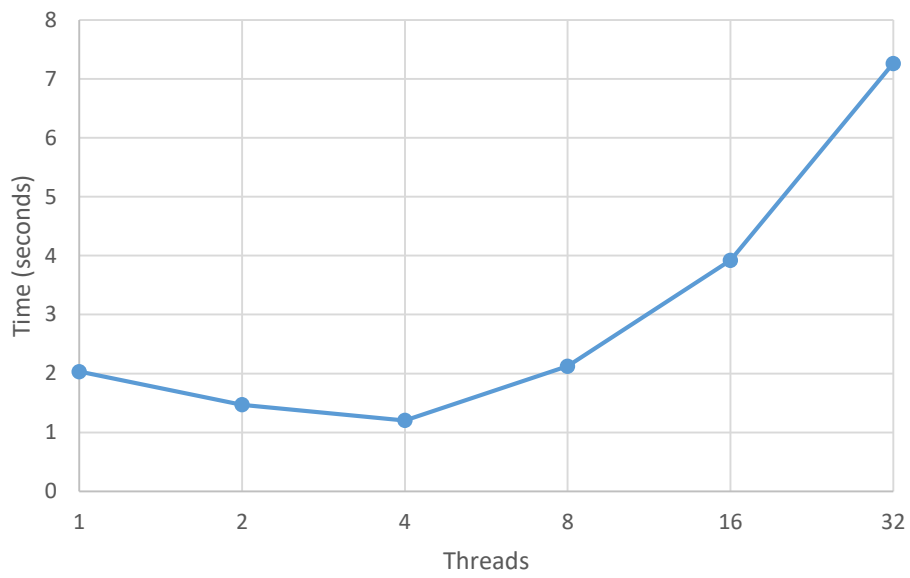
Eksperimentinio tyrimo rezultatai

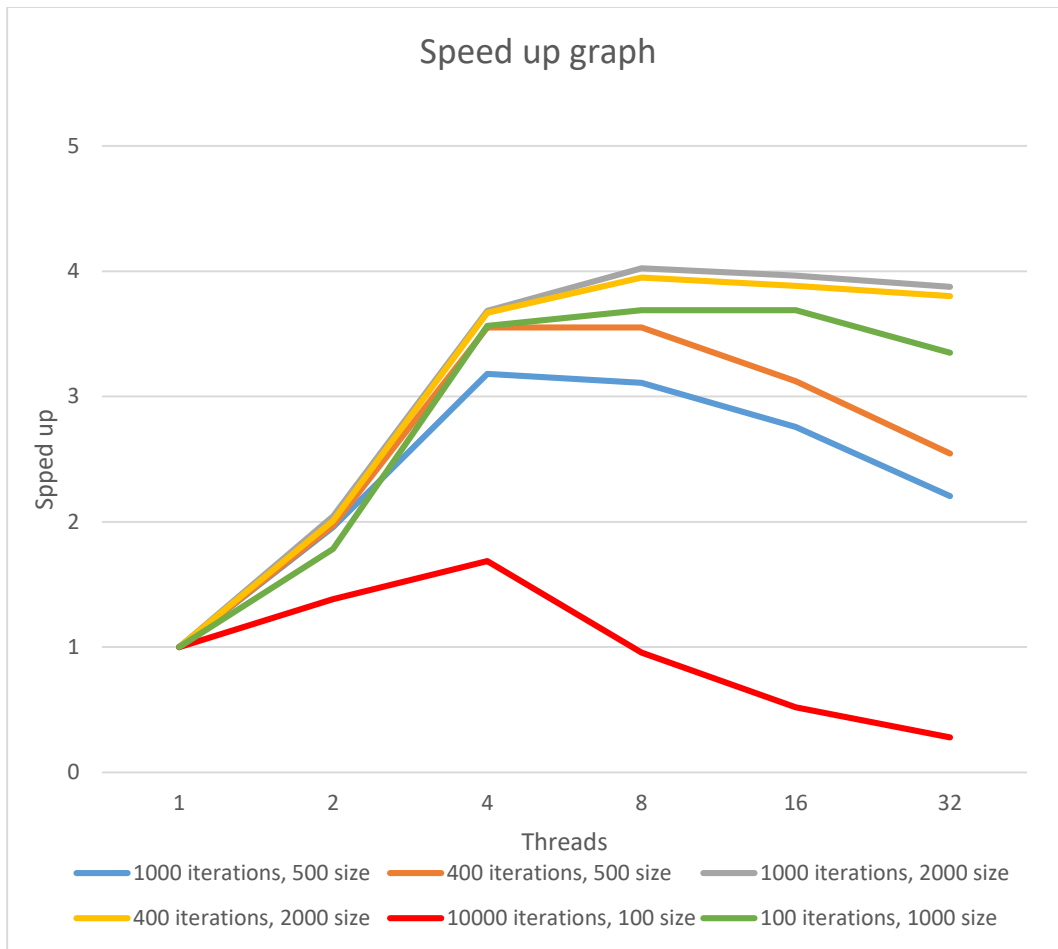


Work load: 1000 iterations, 1000 size



Work load: 10000 iterations, 100 size





Išvados

Darbas smarkiai paspartėja naudojant dvi – keturias gijas. Tačiau ne tiek, plečiant iki 8 gijų. Esant daugiau gijų, darbas pradeda tik lėtėti. Tai natūralu, kadangi naudojamas 8 branduolių procesorius. Naudojant daugiau gijų darbas šiek sulėtėja dėl papildomo darbo su gijų organizavimu bei laukimu barjeruose. Šis sulėtėjimas palyginus nedidelis esant daug duomenų, nes rečiau sustojama barjeruose. Esant daug iteracijų ir mažam objektų kiekiui, spartinimas smarkiai sumažėja.