Flocking simulation (MPI)

***Avtorja:*** Luka Prijatelj in Gašper Kolar.

Kratek opis:

***Dodatne knjižnice***: Tokrat sva za paralelizacijo sva uporabila knjižnico MPI. Za poganjanje na gruči pa sva odstranila knjižnico GLFW ki sva jo do sedaj uporabljala z izris ptic.

***Delitev dela med niti:*** Algoritem potrebuje tabelo ptic kot vhodni podatek. Vsaka ptica je predstavljana z strukturo *Bird*. Ničti procesor pošlje tabelo vseh ptic vsem procesorjem v gruči. Vsak procesor nato izračuna mrežo ter izračuna nove pozicije za svoje ptice. Svoj del poračuna tudi ničti procesor. Nato vsi procesorji pošljejo tabelo ptic z novimi pozicijami ničtemu procesorju. Pošljejo se le ptice ki jim je dani procesor izračunal novo pozicijo. Tako nekoliko prihranimo na komunikaciji. Ničti procesor združi posamezne tabele ptic v skupno tabelo. To skupno tabelo ob naslednji iteraciji spet pošlje vsem procesorjem za izračun novih pozicij.

Če program zaganjamo lokalno je moč vključiti grafični izris. V tem primeru ničti procesor odpre grafično okno in skrbi za izris ptic.

***Komunikacija med nitmi***: Vsaka nit potrebuje podatke o vseh pticah. Ničti procesor tako na začetku vsake iteracije pošlje celotno tabelo ptic na vse procesorje. Ko procesor izračuna nove pozicije ptic pošlje le te nazaj na ničti procesor.

Meritve:

Tabela : Tabela primerja število okvirjev na sekundo za serijski algoritem in paralelni algoritem. FPS predstavlja število okvirjev na sekundo, S predstavlja pohitritev, E pa predstavlja učinkovitost. Meritve so bile napravljene za 2 in 4 MPI niti.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Serial | MPI | | | | | |
| Flock Size | 1 Thread - Grid Optimized | 2 Threads | | | 4 Threads | | |
|  | Frames per second | Frames Per Second [fps] | S2 | E2 | Frames Per Second [fps] | S4 | E4 |
| 1000 | 419.1160 | 386.2993 | 0.9217 | 0.46085 | 309.4379 | 0.738311 | 0.184578 |
| 1200 | 354.7290 | 316.1889 | 0.891353 | 0.445677 | 275.9746 | 0.777987 | 0.194497 |
| 1400 | 296.0220 | 293.1978 | 0.99046 | 0.49523 | 245.1896 | 0.828282 | 0.20707 |
| 1600 | 249.4000 | 245.0158 | 0.982421 | 0.49121 | 218.1782 | 0.874813 | 0.218703 |
| 1800 | 216.0270 | 244.7382 | 1.132906 | 0.566453 | 192.5051 | 0.891116 | 0.222779 |
| 2000 | 186.0880 | 210.0679 | 1.128863 | 0.564432 | 180.1153 | 0.967904 | 0.241976 |
| 2200 | 163.8030 | 171.6193 | 1.047718 | 0.523859 | 166.145 | 1.014298 | 0.253574 |
| 2400 | 143.4850 | 162.3113 | 1.131208 | 0.565604 | 151.8526 | 1.058317 | 0.264579 |
| 2600 | 129.5740 | 138.8439 | 1.071541 | 0.535771 | 139.7364 | 1.078429 | 0.269607 |
| 2800 | 116.0380 | 137.3762 | 1.18389 | 0.591945 | 128.1948 | 1.104765 | 0.276191 |
| AVG: | 227.4282 | 230.5659 | 1.048206 | 0.524103 | 200.733 | 0.933422 | 0.233356 |

Graf : Primerjava števila okvirjev na sekundo v odvisnosti od števila ptic.

Analiza meritev:

**Opis**: Meritve so bile opravljene na osebnem računalniku z dvojedrnim i7 procesorjem. Meritve so bile izvedene z vključenim grafičnim izrisom tako, da so popolnoma primerljive meritvam prejšnjih implementacij. Tako iz kot Graf 1 se vidi da pohitritev hitro pade z večanjem števila niti. Tako kot pri prejšnjih paralelnih implementacija pohitritev in učinkovitost naraščata z večanjem števila ptic a tokrat veliko počasneje. Meritve na štirih nitih so hitreje od serijske implementacije šele pri 2200 pticah.