Programming Languages

TC2006

Juan José Olivera Loyola A01702832

Project Definition

Proposals:

- **Massive Cellular Automata on GPUs**

Game Of Life Experiment

Code: GameOfLife.sni {GoL.cu, gpuGoLSim.cu, cpuGoLSim.cu, mat_utils.h}

We perform Game of Life over a cell matrix of N*N, repeated N_STEPS iterations.
Computation Strategy: Convolutions

<u>System Specs:</u>
Host RAM: 16GB
Host Disk: SSD
Host CPU: Intel Core i5 8300
- Clock Speed: 2.3Ghz [Max 4Ghz]
- Number of Cores: 4 Cores (2 Threads per core)

Device GPU: NVidia GTX 1050
- Clock Speed: 1455 MHz
- RAM: 2GB
- Cores: 640

Experiment 1:
Experiment Parameters:
- N: 1000
- N_STEPS: 1000

1. SubExperiment
    a. Parameters:
        i. ThreadsPerBlock: dim3(32,32)
        ii. BlocksPerGrid: dim(128,128)
    b. Results:
        i. CPU Time: 41.426segs
        ii. GPU Total: 11.245segs
        iii. GPU Iterations: 10.185segs
        iv. Implications:
            1. It takes 10ms to process 1 iteration for the GPU
2. SubExperiment
    a. Parameters:
        i. ThreadsPerBlock: dim3(16,16)
        ii. BlocksPerGrid: dim(128,128)

b. Results:
    i. CPU Time: 40.829segs
    ii. GPU Total: 5.731segs
    iii. GPU Iterations: 4.68segs
    iv. Implications:
        1. It took ~4.68ms to process 1 iteration for the GPU with a quareter the size of maximum available threads poer block
        2. 16x16 = 256

3. SubExperiment
    a. Parameters:
        i. ThreadsPerBlock: dim3(32,16)
        ii. BlocksPerGrid: dim(128,128)
    b. Results:
        i. CPU Time: 40.863segs
        ii. GPU Total: 6.917segs
        iii. GPU Iterations: 5.878segs
        iv. Implications:
            1. It took ~5.6, ~6ms to complete 1 iteration, greater than SubExperiment 2 but less than SubExperiment1
4. SubExperiment
    a. Parameters:
        i. ThreadsPerBlock: dim3(8,8)
        ii. BlocksPerGrid: dim(128,128)
    b. Results:
        i. CPU Time: ----
        ii. GPU Total: 3.961segs
        iii. GPU Iterations: 3.833segs
        iv. Implications:
            1. It took ~3.833, ~4ms to complete 1 iteration
            2. Seems that the lesser the #threads per block, greater the performance for 1000x1000 matrices

TODO: Reducing #Threads Per Block increases performance even for 4000x4000 case??
TODO: Does Mantaining balance has same effects? Performance ~= BLOCKSXTHREADS_PER_BLOCK for some size?
TODO: Do reducing the number of threads while increasing thread internal iteration results in an improvement?

Experiment 2
Experiment Parameters:
- N: 4000
- N_STEPS: 1000

1. SubExperiment
    a. Parameters:

i. ThreadsPerBlock dim3(16,16)
ii. BlocksPerGrid dim3(128,128)

b. Results:
i. CPU Time: 779.993
ii. GPU Total:59.759
iii. GPU Iteration: 58.573
iv. Implications:

Bibliography

[]
https://web.archive.org/web/20100718140020/http://www.swarm.org/images/1/10/How-to_set_up_%26_use_Eclipse_with_Mason.pdf