Griffin Maxwell
9/26/17
CSE 5441

Programming Assignment 1 – Results and Summary

Results at AFFECT_RATE=0.15 and EPSILON=0.15:

These values for `AFFECT_RATE` and `EPSILON` were chosen to try to get the total runtime with `testgrid_400_12206` to be 3-6 minutes. The console output is below:

```
[maxwell.362@beta CSE5441-AMR]$ time ./amr 0.15 0.15 <
/class/cse5441/testgrid_400_12206

*****************************************************************************
temperature dissipation converged in 43234 iterations
    with max DSV = 0.088556 and min DSV = 0.075273
    AFFECT_RATE = 0.150000;   EPSILON = 0.150000
    Num boxes = 12206;  Num rows = 400;   Num columns = 400

elaspsed convergence loop time:
    using clock():          277260000 clicks (277.260010 s)
    using time():           279 s
    using clock_gettime():  278701.000 ms
*****************************************************************************

real  4m38.793s
user  4m37.086s
sys   0m0.259s
```

Results at AFFECT_RATE=0.1 and EPSILON=0.1:

The program was also tested with `AFFECT_RATE=0.1` and `EPSILON=0.1` against all of the `testgrid*` files to compare the how many iterations it took this program to converge with the values provided in the assignment document.

testgrid_1 (? iterations):

```
[maxwell.362@beta CSE5441-AMR]$ time ./amr 0.1 0.1 < /class/cse5441/testgrid_1

*****************************************************************************
temperature dissipation converged in 52 iterations
    with max DSV = 118.918450 and min DSV = 107.278672
    AFFECT_RATE = 0.100000;   EPSILON = 0.100000
    Num boxes = 9;      Num rows = 3;     Num columns = 3
```

```
elaspsed convergence loop time:
    using clock():          0 clicks (0.000000 s)
    using time():           0 s
    using clock_gettime():  0.000 ms
*******************************************************************************

real  0m0.004s
user  0m0.001s
sys   0m0.002s
```

testgrid_2 (245 iterations):

```
[maxwell.362@epsilon cse5441_lab1]$ time ./amr 0.1 0.1 < /class/cse5441/testgrid_2

*******************************************************************************
temperature dissipation converged in 245 iterations
    with max DSV = 55.835885 and min DSV = 50.266851
    AFFECT_RATE = 0.100000;   EPSILON = 0.100000
    Num boxes = 48;     Num rows = 16;    Num columns = 16

elaspsed convergence loop time:
    using clock():          0 clicks (0.000000 s)
    using time():           0 s
    using clock_gettime():  4.000 ms
*******************************************************************************

real  0m0.019s
user  0m0.005s
sys   0m0.003s
```

testgrid_50_78 (1,508 iterations):

```
[maxwell.362@beta CSE5441-AMR]$ time ./amr 0.1 0.1 < /class/cse5441/testgrid_50_78

*******************************************************************************
temperature dissipation converged in 1508 iterations
    with max DSV = 23.369508 and min DSV = 21.035843
    AFFECT_RATE = 0.100000;   EPSILON = 0.100000
    Num boxes = 78;     Num rows = 50;    Num columns = 50

elaspsed convergence loop time:
    using clock():          50000 clicks (0.050000 s)
    using time():           0 s
    using clock_gettime():  53.000 ms
*******************************************************************************

real  0m0.099s
user  0m0.052s
sys   0m0.004s
```

testgrid_50_201 (2,286 iterations):

```
[maxwell.362@beta CSE5441-AMR]$ time ./amr 0.1 0.1 < /class/cse5441/testgrid_50_201

********************************************************************************
temperature dissipation converged in 2286 iterations
    with max DSV = 4.788754 and min DSV = 4.309887
    AFFECT_RATE = 0.100000;   EPSILON = 0.100000
    Num boxes = 201;    Num rows = 50;    Num columns = 50

elaspsed convergence loop time:
    using clock():          150000 clicks (0.150000 s)
    using time():           0 s
    using clock_gettime():  166.000 ms
********************************************************************************

real  0m0.186s
user  0m0.156s
sys   0m0.003s
```

testgrid_200_1166 (14,461 iterations):

```
[maxwell.362@beta CSE5441-AMR]$ time ./amr 0.1 0.1 < /class/cse5441/testgrid_200_1166

********************************************************************************
temperature dissipation converged in 14458 iterations
    with max DSV = 0.812728 and min DSV = 0.731459
    AFFECT_RATE = 0.100000;   EPSILON = 0.100000
    Num boxes = 1166;   Num rows = 200;   Num columns = 200

elaspsed convergence loop time:
    using clock():          5310000 clicks (5.310000 s)
    using time():           5 s
    using clock_gettime():  5309.000 ms
********************************************************************************

real  0m5.322s
user  0m5.316s
sys   0m0.001s
```

testgrid_400_1636 (22,283 iterations):

```
[maxwell.362@beta CSE5441-AMR]$ time ./amr 0.1 0.1 < /class/cse5441/testgrid_400_1636

********************************************************************************
temperature dissipation converged in 22280 iterations
    with max DSV = 1.181786 and min DSV = 1.063610
    AFFECT_RATE = 0.100000;   EPSILON = 0.100000
    Num boxes = 1636;   Num rows = 400;   Num columns = 400

elaspsed convergence loop time:
```

```
    using clock():          11680000 clicks (11.680000 s)
    using time():           12 s
    using clock_gettime():  11684.000 ms
*******************************************************************************

real  0m11.698s
user  0m11.692s
sys   0m0.001s
```

testgrid_400_12206 (75,269 iterations):

```
[maxwell.362@beta CSE5441-AMR]$ time ./amr 0.1 0.1 < /class/cse5441/testgrid_400_12206

*******************************************************************************
temperature dissipation converged in 75197 iterations
    with max DSV = 0.086671 and min DSV = 0.078004
    AFFECT_RATE = 0.100000;   EPSILON = 0.100000
    Num boxes = 12206;  Num rows = 400;   Num columns = 400

elaspsed convergence loop time:
    using clock():          436500000 clicks (436.500000 s)
    using time():           437 s
    using clock_gettime():  436914.000 ms
*******************************************************************************

real  7m16.986s
user  7m16.435s
sys   0m0.122s
```

Summary of Timing Results:

Test Grid File	Affect Rate	Epsilon	"Real" Time to Converge	"clock()" Time to Converge	Expected Iterations to Converge	Actual Iterations to Converge	% error
testgrid_400_12206	0.15	0.15	4m38.793s	277.26 s	-	43,234	-
testgrid_1	0.1	0.1	0m0.004s	0.00 s	-	52	-
testgrid_2	0.1	0.1	0m0.019s	0.00 s	245	245	0%
testgrid_50_78	0.1	0.1	0m0.099s	0.05 s	1,508	1,508	0%
testgrid_50_201	0.1	0.1	0m0.186s	0.15 s	2,286	2,286	0%
testgrid_200_1166	0.1	0.1	0m5.322s	5.31 s	14,461	14,458	0.021%
testgrid_400_1636	0.1	0.1	0m11.698s	11.68 s	22,283	22,280	0.013%
testgrid_400_12206	0.1	0.1	7m16.986s	436.50 s	75,269	75,197	0.096%

The timing and convergence results are summarized in the table above. As can be seen by the % error column, the number of iterations for my algorithm to converge on each test grid is extremely close to those given in the assignment document. This give me high confidence that my dissipation model very closely matches the one given in the assignment document, which was the goal. These small differences are likely caused by the order in which all the double-precision floating point calculations are done. Even if the error between any of my DSV calculations and the ones used to generate the given numbers is miniscule, the small error can compound over a large number of calculations and a large number of iterations. This is evidenced by the fact that the % error seems to grow as the grid size and number of boxes on the grid grows.

Additionally, the timings of all my runs were fairly reasonable. In order to get a run with the largest grid to last between 3 and 6 minutes, the original values of affect rate and epsilon didn't have to change too much; affect rate = 0.15 and epsilon = 0.15 makes the program converge in about 4.5 minutes. Furthermore, the time it took to run each test grid grew exponentially as the grid and number of boxes increased. This makes sense because more boxes mean more potential neighbors, which means longer lists of neighbors, which leads to more number crunching to calculate the weighted average adjacent temperature per cell. This, on top of the fact that there are more cells to update per iteration to begin with, is what makes the execution time grow so rapidly. Since this program is currently serial, it can't gain any of the benefits of parallel processes updating more than one cell at a time, which makes me predict that a more parallelized program will run much faster, and heavily reduce how harsh this exponential growth is.