Report for a2

Sun Can

31309202

Run on node2x14a

Exercise 1

1. There are no data races in sssp\_init() if we separate the loop into sub-loops. There is no overlap between data in the loop.
2. Data race in sssp\_round() is the <changed> flag value and g.node\_wt weight of each node.

I separate the changed flag to each thread and see if the flag is on after the synchronization of every thread.

1. For the sssp:

|  |  |  |  |
| --- | --- | --- | --- |
| 4 threads | 8 threads | 16 threads |  |
| 1203 | 867 | 682 |  |
| 1193 | 839 | 740 |  |
| 1212 | 861 | 734 |  |
| 1217 | 851 | 683 |  |
| 1204 | 827 | 685 |  |
| 1205.8 | 849 | 704.8 | AVERAGE |
| 9.20326 | 16.2480 | 29.4906 | STD |
|  |  |  |  |

For the sssp\_barrier:

|  |  |  |  |
| --- | --- | --- | --- |
| 4 threads | 8 threads | 16 threads |  |
| 1275 | 924 | 759 |  |
| 1291 | 915 | 772 |  |
| 1278 | 947 | 759 |  |
| 1275 | 909 | 781 |  |
| 1287 | 977 | 778 |  |
| 1281.2 | 934.4 | 769.8 | AVG |
| 7.36206493 | 27.8531865 | 10.3778611 | STD |

Actually, on my own computer it seems that barrier one is quicker but, on the machine, it seems to be a little bit slower I guess it is because of the overhead.

Exercise 2



|  |  |  |  |
| --- | --- | --- | --- |
| 4 threads | 8 threads | 16 threads |  |
| 3383 | 3491 | 3630 |  |
| 3390 | 3486 | 3638 |  |
| 3388 | 3481 | 3626 |  |
| 3392 | 3492 | 3639 |  |
| 3385 | 3490 | 3638 |  |
| 3387.6 | 3488 | 3634.2 | AVERAGE |
| 3.64691651 | 4.52769257 | 5.84807661 | STD |

The overhead is great and as number of threads increases, the cost goes up.

1. I got deadlock on big number of threads and the strategy is to relax the load part of data and lock the write of ‘wt’ variables.

The deadlock is mainly caused by the overlap of the two lock of starting node and ending node of an edge. When it is going to visit the starting node, it has been blocked because others are entering the lock and being stuck by the ending lock.

1. If we do not use any lock, it will soon end in the middle of program and leave many nodes INF and I believe the semantics of SSSP does not hold in this situation.

Exercise 3

[csun18@node2x14a src]$ ./sssp ./inputs/graphs/NY.txt ./my-outputs/NY-s.txt 4

Loaded './inputs/graphs/NY.txt', 264346 nodes, 730100 edges

612 rounds

Total time: 1203 ms

Thread 0 time 513

Thread 1 time 537

Thread 2 time 547

Thread 3 time 511

Wrote output './my-outputs/NY-s.txt'

[csun18@node2x14a src]$ ./sssp ./inputs/graphs/NY.txt ./my-outputs/NY-s.txt 4

Loaded './inputs/graphs/NY.txt', 264346 nodes, 730100 edges

612 rounds

Total time: 1193 ms

Thread 0 time 486

Thread 1 time 537

Thread 2 time 510

Thread 3 time 474

Wrote output './my-outputs/NY-s.txt'

[csun18@node2x14a src]$ ./sssp ./inputs/graphs/NY.txt ./my-outputs/NY-s.txt 4

Loaded './inputs/graphs/NY.txt', 264346 nodes, 730100 edges

612 rounds

Total time: 1212 ms

Thread 0 time 526

Thread 1 time 573

Thread 2 time 556

Thread 3 time 459

Wrote output './my-outputs/NY-s.txt'

[csun18@node2x14a src]$ ./sssp ./inputs/graphs/NY.txt ./my-outputs/NY-s.txt 4

Loaded './inputs/graphs/NY.txt', 264346 nodes, 730100 edges

612 rounds

Total time: 1217 ms

Thread 0 time 553

Thread 1 time 547

Thread 2 time 543

Thread 3 time 451

Wrote output './my-outputs/NY-s.txt'

[csun18@node2x14a src]$ ./sssp ./inputs/graphs/NY.txt ./my-outputs/NY-s.txt 4

Loaded './inputs/graphs/NY.txt', 264346 nodes, 730100 edges

612 rounds

Total time: 1204 ms

Thread 0 time 483

Thread 1 time 553

Thread 2 time 558

Thread 3 time 529

Wrote output './my-outputs/NY-s.txt'

[csun18@node2x14a src]$ ./sssp ./inputs/graphs/NY.txt ./my-outputs/NY-s.txt 4

Loaded './inputs/graphs/NY.txt', 264346 nodes, 730100 edges

612 rounds

Total time: 1221 ms

Thread 0 time 537

Thread 1 time 551

Thread 2 time 556

Thread 3 time 504

Wrote output './my-outputs/NY-s.txt'

[csun18@node2x14a src]$ ./sssp ./inputs/graphs/NY.txt ./my-outputs/NY-s.txt 8

Loaded './inputs/graphs/NY.txt', 264346 nodes, 730100 edges

612 rounds

Total time: 867 ms

Thread 0 time 62

Thread 1 time 241

Thread 2 time 239

Thread 3 time 244

Thread 4 time 150

Thread 5 time 123

Thread 6 time 196

Thread 7 time 48

Wrote output './my-outputs/NY-s.txt'

[csun18@node2x14a src]$ ./sssp ./inputs/graphs/NY.txt ./my-outputs/NY-s.txt 8

Loaded './inputs/graphs/NY.txt', 264346 nodes, 730100 edges

612 rounds

Total time: 839 ms

Thread 0 time 84

Thread 1 time 234

Thread 2 time 250

Thread 3 time 233

Thread 4 time 141

Thread 5 time 108

Thread 6 time 89

Thread 7 time 43

Wrote output './my-outputs/NY-s.txt'

[csun18@node2x14a src]$ ./sssp ./inputs/graphs/NY.txt ./my-outputs/NY-s.txt 8

Loaded './inputs/graphs/NY.txt', 264346 nodes, 730100 edges

611 rounds

Total time: 861 ms

Thread 0 time 68

Thread 1 time 240

Thread 2 time 245

Thread 3 time 248

Thread 4 time 148

Thread 5 time 118

Thread 6 time 153

Thread 7 time 45

Wrote output './my-outputs/NY-s.txt'

[csun18@node2x14a src]$ ./sssp ./inputs/graphs/NY.txt ./my-outputs/NY-s.txt 8

Loaded './inputs/graphs/NY.txt', 264346 nodes, 730100 edges

612 rounds

Total time: 851 ms

Thread 0 time 59

Thread 1 time 218

Thread 2 time 247

Thread 3 time 231

Thread 4 time 148

Thread 5 time 111

Thread 6 time 149

Thread 7 time 45

Wrote output './my-outputs/NY-s.txt'

[csun18@node2x14a src]$ ./sssp ./inputs/graphs/NY.txt ./my-outputs/NY-s.txt 8

Loaded './inputs/graphs/NY.txt', 264346 nodes, 730100 edges

613 rounds

Total time: 827 ms

Thread 0 time 62

Thread 1 time 139

Thread 2 time 234

Thread 3 time 225

Thread 4 time 151

Thread 5 time 123

Thread 6 time 163

Thread 7 time 39

Wrote output './my-outputs/NY-s.txt'

[csun18@node2x14a src]$ ./sssp ./inputs/graphs/NY.txt ./my-outputs/NY-s.txt 16

Loaded './inputs/graphs/NY.txt', 264346 nodes, 730100 edges

612 rounds

Total time: 682 ms

Thread 0 time 0

Thread 1 time 2

Thread 2 time 0

Thread 3 time 3

Thread 4 time 83

Thread 5 time 68

Thread 6 time 24

Thread 7 time 66

Thread 8 time 70

Thread 9 time 56

Thread 10 time 53

Thread 11 time 26

Thread 12 time 0

Thread 13 time 0

Thread 14 time 0

Thread 15 time 0

Here is the report of 4-16 threads.

Under the situation of 4 threads there are no significant imbalance.

For 8 and 16 threads, we can see from above that the middle part of threads is having heavier loads than the starting and ending threads. That is because the loop of threads will not reach that many rounds and the problem will be solved already by other threads or at the beginning the map is not that complicate that it has not that many conditional variables to calculate.