1.2 Competitor Analysis (20%)

Through equally shady means, your competitors, SaaS-Whole, have acquired this same user matrix, and your definitely illegal corporate espionage team estimates that SaaS-Whole will finish solving their equivalent problem in 3 days.

a) Assume that the sequential version of your program is 99.9% parallelizable. You estimate that the runtime without any parallelism would be 300,000,000 seconds. Do you have any hope of finishing the computation before your competitors? What if the estimated runtime were 150,000,000 seconds?

$$300{,}000{,}000 \ \ seconds \ = \ 83{,}333.3 \ \ hours \ = \ 3{,}472.2 \ \ days$$

$$Required \ \ Speedup \ = \ \frac{300{,}000{,}000 \ \ seconds}{259{,}200 \ \ seconds} \ = \ 1{,}157.41$$

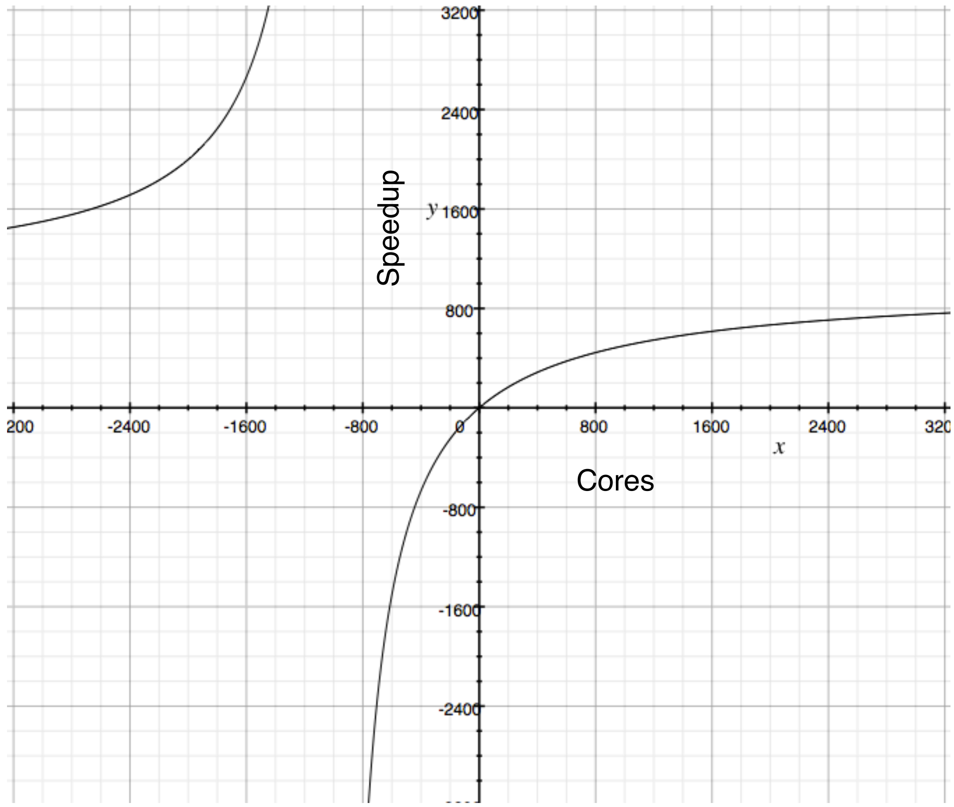
$$S(n) \ = \ \frac{1}{1 \ - \ P \ + \ \frac{P}{n}}$$

$$1{,}157.41 \ = \ \frac{1}{0.001 \ + \ \frac{0.999}{n}}$$

$$n \ = \ -7364.64$$

n specifies the number of cores required to achieve the speedup. By plotting the following equation we can see that it will not be possible to achieve the speedup required to finish the computation before the competitors.

$$S(n) \ = \ \frac{1}{0.001 \ + \ \frac{0.999}{n}}$$



Clearly, it's impossible to achieve a speedup of 1,157.41; there is no chance we can beat the competition. This is why the n is negative. We now do the same for the runtime of 150,000,000 seconds.

$$150{,}000{,}000 \ \ seconds \ = \ 41{,}666.67 \ \ hours \ = \ 1{,}736.1 \ \ days$$

$$Required \ \ Speedup \ = \ \frac{150{,}000{,}000 \ \ seconds}{259{,}200 \ \ seconds} \ = \ 578.70$$

$$578.70 \ = \ \frac{1}{0.001 \ + \ \frac{0.999}{n}}$$

$$n \ = \ 1372.25$$

In this scenario it will be possible to beat the competition. As the graph above shows, a speedup of 578.70 can be achieved using 1372 cores.

b) Repeat a. using the constrained version of the Amdahl bound that you derived in Theoretical Assignment 2, assuming that OV(n) = 10n. (You can either use calculus to find the answer analytically or write a program to find the answer numerically.)

$$S(n) = \frac{T_{seq}}{T_{parallel}} = \frac{T_{seq}}{T_{seq}(1 - P) + T_{seq}(\frac{P}{n}) + 10n}$$

$$S(n) = \frac{300{,}000{,}000}{300{,}000{,}000 \times 0.001 + 300{,}000{,}000 \times \frac{0.999}{n} + 10n}$$

$$First\ derivative\ of\ S\ (n) = \frac{-30{,}000{,}000(n^2 - 29{,}970{,}000)}{(n^2 + 30{,}000n + 29{,}970{,}000)^2}$$

$$0 = \frac{-30{,}000{,}000(n^2 - 29{,}970{,}000)}{(n^2 + 30{,}000Nn + 29{,}970{,}000)^2}$$

$$n = 900\sqrt{37} = 5{,}475.49$$

$$S(5476) = \frac{300{,}000{,}000}{300{,}000{,}000 \times 0.001 + 300{,}000{,}000 \times \frac{0.999}{5476} + 10 \times 5476} = 732.62$$

As it was calculated in part a, the required speedup to beat the competitors is 1,157.41. The maximum speedup that can be achieved with overhead 10n is 732.62. There is no hope of finishing the computation before the competitor when the sequential execution time is 300,000,000 seconds. Next, we perform the same calculation for a sequential execution time of 150,000,000 seconds.

As it was calculated in part a, the required speedup to beat the competitors is 578.7037037. The maximum speedup that can be achieved with overhead 10n in this scenario is 659.56. There is hope of finishing the computation before the competitor when the sequential execution time is 150,000,000 seconds.

$$S(n) = \frac{T_{seq}}{T_{parallel}} = \frac{T_{seq}}{T_{seq}(1 - P) + T_{seq}(\frac{P}{n}) + 10n}$$

$$S(n) = \frac{150{,}000{,}000}{150{,}000{,}000 \times 0.001 + 150{,}000{,}000 \times \frac{0.999}{n} + 10n}$$

$$First\ derivative\ of\ S\ (n) = \frac{-15{,}000{,}000(n^2 - 14{,}985{,}000)}{(n^2 + 15{,}000n + 14{,}985{,}000)^2}$$

$$0 = \frac{-15{,}000{,}000(n^2 - 14{,}985{,}000)}{(n^2 + 15{,}000n + 14{,}985{,}000)^2}$$

$$n = 450\sqrt{74} = 3{,}871.05$$

$$S(3871) = \frac{150{,}000{,}000}{150{,}000{,}000 \times 0.001 + 150{,}000{,}000 \times \frac{0.999}{3871} + 10 \times 3871} = 659.56$$

c) Estimate what percentage of your program's runtime is parallelizable using the stopwatch commands provided in Programming Assignment 1 or some other timing method. In your report, explain how your measurement works and justify your method. Repeat a. using your program’s percentage instead of 99.9%.

The parallelization program submitted with this report successfully parallelizes the computation between the processors of the host. However, I made the decision to have a single process read the entire matrix from the file, and distribute its contents between the processors using mpi_scatterv(); the overhead from this step is so large that the sequential algorithm is faster for all matrix sizes that I tested (up to 5000 by 5001 elements). This is disappointing, but it provides a valuable lesson for the future. In order to answer this question, I exclude the matrix read operation from the timing analysis of the parallelized algorithm, as well as the processing for question 1.4, such that only the computational part is analyzed.

After executing the parallelized algorithm one hundred times on a 2000 by 2001 elements matrix, we find that the average execution time on 4 processes is 15,503,651 microseconds. Doing the same for the sequential version of the algorithm showed an average execution time of 17,805,571 microseconds.

Using these execution times, we calculate the average speedup of the program when executed on 4 cores:

$$S(4) = \frac{17{,}805{,}571}{15{,}503{,}651} = 1.148$$

Using Ahmdal's Law we calculate the fraction of the program that is parallelizable.

$$S(n) = \frac{1}{(1 - P) + \frac{P}{n}}$$

$$S(n)(1 - P) + S(n)\frac{P}{n} = 1$$

$$P = \frac{1 - S(n)}{\frac{S(n)}{n} - S(n)}$$

$$P = \frac{1 - 1.148}{\frac{1.148}{4} - 1.148} = 0.172$$

Thus, it can be seen that the percentage of the program that can be parallelized is 17.2%. Next we find the execution time with this new value for P using 300,000,000 seconds as the sequential execution time.

$$T(n) = 300{,}000{,}000((1 - P) + \frac{P}{n})$$

$$T(n) = 300{,}000{,}000((1 - 0.172) + \frac{0.172}{n})$$

$$\lim_{n \to \infty} T(n) = 300{,}000{,}000(1 - 0.172) = 248{,}400{,}000 \ seconds = 2{,}875 \ days$$

The fastest time in which the processing can be completed with an algorithm that is 17.2% parallelizable is 2,875 days. Given that SaaS-Whole will finish their processing in 3 days, there is no hope of beating them. We now calculate the time considering a sequential execution time of 150,000,000 seconds.

$$T(n) = 150{,}000{,}000((1 - 0.172) + \frac{0.172}{n})$$

$$\lim_{n \to \infty} T(n) = 150{,}000{,}000(1 - 0.172) = 124{,}200{,}000 \ seconds = 1437.5 \ days$$

In this case, the fastest processing time is 1437.5 days. We have no hope of defeating SaaS-Whole.

1.3 Deployment (20%)

You decide to deploy your code on a large cluster computer. Your IaaS provider, Nebulous Contract, charges $0.01 per compute-hour (i.e., the cost of running two CPU instances for one hour is $0.02). Assuming once more that your program is 99.9% parallelizable and takes 150,000,000 seconds to run, using the constrained Amdahl bound, how much would it cost to buy enough CPU instances from Nebulous Contract to run your MPI program and achieve i) half and ii) three-quarters of the theoretical maximum speedup?

i) Half of the theoretical maximum speedup

The theoretical maximum speedup was calculated in part B of question 1.2. It's value is 659.56. The application needs to be executed on 3,871 processors in order to achieve this speedup.

We need to calculate the number of processors needed to achieve half of the maximum speedup. Half of the theoretical maximum speedup is 329.78.

$$S(n) = \frac{150{,}000{,}000}{150{,}000{,}000 \times 0.001 + 150{,}000{,}000 \times \frac{0.999}{n} + 10n}$$

$$329.78 = \frac{150{,}000{,}000}{150{,}000{,}000 \times 0.001 + 150{,}000{,}000 \times \frac{0.999}{n} + 10n}$$

$$n = 499.75 \approx 500$$

$$S(n) = 329.78 = \frac{150{,}000{,}000}{T_{parallel}}$$

$$T_{parallel} = 454{,}848.687 \ \ seconds = 126.35 \ \ hours$$

$$Total \ \ Cost = 126.35 \ \ hours \times 500 \ \ CPUs \times \$0.01 = \$631.42$$

ii) Three-quarters of the theoretical maximum speedup

We need to calculate the number of processors needed to achieve three-quarters of the maximum speedup. Three-quarters of the theoretical maximum speedup is 494.67.

$$494.67 = \frac{150{,}000{,}000}{150{,}000{,}000 \times 0.001 + 150{,}000{,}000 \times \frac{0.999}{n} + 10n}$$

$$n = 1049.86 \approx 1050$$

$$S(n) = 494.67 = \frac{150{,}000{,}000}{T_{parallel}}$$

$$T_{parallel} = 303{,}232.458 \ \ seconds = 84.23 \ \ hours$$

$$Total \ \ Cost = 84.23 \ \ hours \times 1050 \ \ CPUs \times \$0.01 = \$884.31$$

Bonus Question

I executed my program on 3 hosts available from ECE. As specified on the McGill ECE FAQ, the available machines are TR5130GU-1.ECE.McGill.CA to TR5130GU-15.ECE.McGill.CA. I ran my program on ECE12, ECE13, and ECE10 as can be seen in the debug log below. I am certain that my program produces the correct output. For testing, I used a matrix of 2000 by 2001 elements.

```
[daniel.macario@mail.mcgill.ca@tr5130gu-13 ~/Documents/FALL_2015/ECSE_420/
assignmentSolutions/A4_2]$ mpirun -H ECE12,ECE13,ECE10 -np 8 -d ./a.out mat_gen_test.txt
[tr5130gu-13.ece.mcgill.ca:15668] procdir: /tmp/openmpi-sessions-
daniel.macario@mail.mcgill.ca@tr5130gu-13.ece.mcgill.ca_0/22357/0/0
[tr5130gu-13.ece.mcgill.ca:15668] jobdir: /tmp/openmpi-sessions-
daniel.macario@mail.mcgill.ca@tr5130gu-13.ece.mcgill.ca_0/22357/0
[tr5130gu-13.ece.mcgill.ca:15668] top: openmpi-sessions-
daniel.macario@mail.mcgill.ca@tr5130gu-13.ece.mcgill.ca_0
[tr5130gu-13.ece.mcgill.ca:15668] tmp: /tmp
[tr5130gu-13.ece.mcgill.ca:15675] procdir: /tmp/openmpi-sessions-
daniel.macario@mail.mcgill.ca@tr5130gu-13.ece.mcgill.ca_0/22357/0/2
[tr5130gu-13.ece.mcgill.ca:15675] jobdir: /tmp/openmpi-sessions-
daniel.macario@mail.mcgill.ca@tr5130gu-13.ece.mcgill.ca_0/22357/0
[tr5130gu-13.ece.mcgill.ca:15675] top: openmpi-sessions-
daniel.macario@mail.mcgill.ca@tr5130gu-13.ece.mcgill.ca_0
[tr5130gu-13.ece.mcgill.ca:15675] tmp: /tmp
[tr5130gu-12.ece.mcgill.ca:02392] procdir: /tmp/openmpi-sessions-
daniel.macario@mail.mcgill.ca@tr5130gu-12.ece.mcgill.ca_0/22357/0/1
[tr5130gu-12.ece.mcgill.ca:02392] jobdir: /tmp/openmpi-sessions-
daniel.macario@mail.mcgill.ca@tr5130gu-12.ece.mcgill.ca_0/22357/0
[tr5130gu-12.ece.mcgill.ca:02392] top: openmpi-sessions-
daniel.macario@mail.mcgill.ca@tr5130gu-12.ece.mcgill.ca_0
[tr5130gu-12.ece.mcgill.ca:02392] tmp: /tmp
[tr5130gu-10.ece.mcgill.ca:03611] procdir: /tmp/openmpi-sessions-
daniel.macario@mail.mcgill.ca@tr5130gu-10.ece.mcgill.ca_0/22357/0/3
[tr5130gu-10.ece.mcgill.ca:03611] jobdir: /tmp/openmpi-sessions-
daniel.macario@mail.mcgill.ca@tr5130gu-10.ece.mcgill.ca_0/22357/0
[tr5130gu-10.ece.mcgill.ca:03611] top: openmpi-sessions-
daniel.macario@mail.mcgill.ca@tr5130gu-10.ece.mcgill.ca_0
[tr5130gu-10.ece.mcgill.ca:03611] tmp: /tmp
[tr5130gu-13.ece.mcgill.ca:15675] [[22357,0],2] node[0].name tr5130gu-13 daemon 0
[tr5130gu-13.ece.mcgill.ca:15675] [[22357,0],2] node[1].name ECE12 daemon 1
[tr5130gu-13.ece.mcgill.ca:15675] [[22357,0],2] node[2].name ECE13 daemon 2
[tr5130gu-13.ece.mcgill.ca:15675] [[22357,0],2] node[3].name ECE10 daemon 3
[tr5130gu-12.ece.mcgill.ca:02392] [[22357,0],1] node[0].name tr5130gu-13 daemon 0
[tr5130gu-12.ece.mcgill.ca:02392] [[22357,0],1] node[1].name ECE12 daemon 1
[tr5130gu-12.ece.mcgill.ca:02392] [[22357,0],1] node[2].name ECE13 daemon 2
[tr5130gu-12.ece.mcgill.ca:02392] [[22357,0],1] node[3].name ECE10 daemon 3
[tr5130gu-10.ece.mcgill.ca:03611] [[22357,0],3] node[0].name tr5130gu-13 daemon 0
[tr5130gu-10.ece.mcgill.ca:03611] [[22357,0],3] node[1].name ECE12 daemon 1
[tr5130gu-10.ece.mcgill.ca:03611] [[22357,0],3] node[2].name ECE13 daemon 2
[tr5130gu-10.ece.mcgill.ca:03611] [[22357,0],3] node[3].name ECE10 daemon 3
[tr5130gu-13.ece.mcgill.ca:15676] procdir: /tmp/openmpi-sessions-
daniel.macario@mail.mcgill.ca@tr5130gu-13.ece.mcgill.ca_0/22357/1/1
[tr5130gu-13.ece.mcgill.ca:15676] jobdir: /tmp/openmpi-sessions-
daniel.macario@mail.mcgill.ca@tr5130gu-13.ece.mcgill.ca_0/22357/1
```

```
[tr5130gu-13.ece.mcgill.ca:15676] top: openmpi-sessions-
daniel.macario@mail.mcgill.ca@tr5130gu-13.ece.mcgill.ca_0
[tr5130gu-13.ece.mcgill.ca:15676] tmp: /tmp
[tr5130gu-13.ece.mcgill.ca:15676] [[22357,1],1] node[0].name tr5130gu-13 daemon 0
[tr5130gu-13.ece.mcgill.ca:15676] [[22357,1],1] node[1].name ECE12 daemon 1
[tr5130gu-13.ece.mcgill.ca:15676] [[22357,1],1] node[2].name ECE13 daemon 2
[tr5130gu-13.ece.mcgill.ca:15676] [[22357,1],1] node[3].name ECE10 daemon 3
[tr5130gu-13.ece.mcgill.ca:15677] procdir: /tmp/openmpi-sessions-
daniel.macario@mail.mcgill.ca@tr5130gu-13.ece.mcgill.ca_0/22357/1/4
[tr5130gu-13.ece.mcgill.ca:15677] jobdir: /tmp/openmpi-sessions-
daniel.macario@mail.mcgill.ca@tr5130gu-13.ece.mcgill.ca_0/22357/1
[tr5130gu-13.ece.mcgill.ca:15677] top: openmpi-sessions-
daniel.macario@mail.mcgill.ca@tr5130gu-13.ece.mcgill.ca_0
[tr5130gu-13.ece.mcgill.ca:15677] tmp: /tmp
[tr5130gu-13.ece.mcgill.ca:15677] [[22357,1],4] node[0].name tr5130gu-13 daemon 0
[tr5130gu-13.ece.mcgill.ca:15677] [[22357,1],4] node[1].name ECE12 daemon 1
[tr5130gu-13.ece.mcgill.ca:15677] [[22357,1],4] node[2].name ECE13 daemon 2
[tr5130gu-13.ece.mcgill.ca:15677] [[22357,1],4] node[3].name ECE10 daemon 3
[tr5130gu-13.ece.mcgill.ca:15678] procdir: /tmp/openmpi-sessions-
daniel.macario@mail.mcgill.ca@tr5130gu-13.ece.mcgill.ca_0/22357/1/7
[tr5130gu-13.ece.mcgill.ca:15678] jobdir: /tmp/openmpi-sessions-
daniel.macario@mail.mcgill.ca@tr5130gu-13.ece.mcgill.ca_0/22357/1
[tr5130gu-13.ece.mcgill.ca:15678] top: openmpi-sessions-
daniel.macario@mail.mcgill.ca@tr5130gu-13.ece.mcgill.ca_0
[tr5130gu-13.ece.mcgill.ca:15678] tmp: /tmp
[tr5130gu-13.ece.mcgill.ca:15678] [[22357,1],7] node[0].name tr5130gu-13 daemon 0
[tr5130gu-13.ece.mcgill.ca:15678] [[22357,1],7] node[1].name ECE12 daemon 1
[tr5130gu-13.ece.mcgill.ca:15678] [[22357,1],7] node[2].name ECE13 daemon 2
[tr5130gu-13.ece.mcgill.ca:15678] [[22357,1],7] node[3].name ECE10 daemon 3
 MPIR_being_debugged = 0
 MPIR_debug_state = 1
 MPIR_partial_attach_ok = 1
 MPIR_i_am_starter = 0
 MPIR_forward_output = 0
 MPIR_proctable_size = 8
 MPIR_proctable:
   (i, host, exe, pid) = (0, ECE12, /home/dmacar2/Documents/FALL_2015/ECSE_420/
assignmentSolutions/A4_2/./a.out, 2393)
   (i, host, exe, pid) = (1, ECE13, /home/dmacar2/Documents/FALL_2015/ECSE_420/
assignmentSolutions/A4_2/./a.out, 15676)
   (i, host, exe, pid) = (2, ECE10, /home/dmacar2/Documents/FALL_2015/ECSE_420/
assignmentSolutions/A4_2/./a.out, 3612)
   (i, host, exe, pid) = (3, ECE12, /home/dmacar2/Documents/FALL_2015/ECSE_420/
assignmentSolutions/A4_2/./a.out, 2394)
   (i, host, exe, pid) = (4, ECE13, /home/dmacar2/Documents/FALL_2015/ECSE_420/
assignmentSolutions/A4_2/./a.out, 15677)
   (i, host, exe, pid) = (5, ECE10, /home/dmacar2/Documents/FALL_2015/ECSE_420/
assignmentSolutions/A4_2/./a.out, 3613)
   (i, host, exe, pid) = (6, ECE12, /home/dmacar2/Documents/FALL_2015/ECSE_420/
assignmentSolutions/A4_2/./a.out, 2395)
   (i, host, exe, pid) = (7, ECE13, /home/dmacar2/Documents/FALL_2015/ECSE_420/
assignmentSolutions/A4_2/./a.out, 15678)
MPIR_executable_path: NULL
MPIR_server_arguments: NULL
```

[tr5130gu-12.ece.mcgill.ca:02393] procdir: /tmp/openmpi-sessions-daniel.macario@mail.mcgill.ca@tr5130gu-12.ece.mcgill.ca_0/22357/1/0
[tr5130gu-12.ece.mcgill.ca:02393] jobdir: /tmp/openmpi-sessions-daniel.macario@mail.mcgill.ca@tr5130gu-12.ece.mcgill.ca_0/22357/1
[tr5130gu-12.ece.mcgill.ca:02393] top: openmpi-sessions-daniel.macario@mail.mcgill.ca@tr5130gu-12.ece.mcgill.ca_0
[tr5130gu-12.ece.mcgill.ca:02393] tmp: /tmp
[tr5130gu-12.ece.mcgill.ca:02393] [[22357,1],0] node[0].name tr5130gu-13 daemon 0
[tr5130gu-12.ece.mcgill.ca:02393] [[22357,1],0] node[1].name ECE12 daemon 1
[tr5130gu-12.ece.mcgill.ca:02393] [[22357,1],0] node[2].name ECE13 daemon 2
[tr5130gu-12.ece.mcgill.ca:02393] [[22357,1],0] node[3].name ECE10 daemon 3
[tr5130gu-12.ece.mcgill.ca:02394] procdir: /tmp/openmpi-sessions-daniel.macario@mail.mcgill.ca@tr5130gu-12.ece.mcgill.ca_0/22357/1/3
[tr5130gu-12.ece.mcgill.ca:02394] jobdir: /tmp/openmpi-sessions-daniel.macario@mail.mcgill.ca@tr5130gu-12.ece.mcgill.ca_0/22357/1
[tr5130gu-12.ece.mcgill.ca:02394] top: openmpi-sessions-daniel.macario@mail.mcgill.ca@tr5130gu-12.ece.mcgill.ca_0
[tr5130gu-12.ece.mcgill.ca:02394] tmp: /tmp
[tr5130gu-12.ece.mcgill.ca:02395] procdir: /tmp/openmpi-sessions-daniel.macario@mail.mcgill.ca@tr5130gu-12.ece.mcgill.ca_0/22357/1/6
[tr5130gu-12.ece.mcgill.ca:02395] jobdir: /tmp/openmpi-sessions-daniel.macario@mail.mcgill.ca@tr5130gu-12.ece.mcgill.ca_0/22357/1
[tr5130gu-12.ece.mcgill.ca:02395] top: openmpi-sessions-daniel.macario@mail.mcgill.ca@tr5130gu-12.ece.mcgill.ca_0
[tr5130gu-12.ece.mcgill.ca:02395] tmp: /tmp
[tr5130gu-12.ece.mcgill.ca:02394] [[22357,1],3] node[0].name tr5130gu-13 daemon 0
[tr5130gu-12.ece.mcgill.ca:02394] [[22357,1],3] node[1].name ECE12 daemon 1
[tr5130gu-12.ece.mcgill.ca:02394] [[22357,1],3] node[2].name ECE13 daemon 2
[tr5130gu-12.ece.mcgill.ca:02394] [[22357,1],3] node[3].name ECE10 daemon 3
[tr5130gu-12.ece.mcgill.ca:02395] [[22357,1],6] node[0].name tr5130gu-13 daemon 0
[tr5130gu-12.ece.mcgill.ca:02395] [[22357,1],6] node[1].name ECE12 daemon 1
[tr5130gu-12.ece.mcgill.ca:02395] [[22357,1],6] node[2].name ECE13 daemon 2
[tr5130gu-12.ece.mcgill.ca:02395] [[22357,1],6] node[3].name ECE10 daemon 3
[tr5130gu-10.ece.mcgill.ca:03612] procdir: /tmp/openmpi-sessions-daniel.macario@mail.mcgill.ca@tr5130gu-10.ece.mcgill.ca_0/22357/1/2
[tr5130gu-10.ece.mcgill.ca:03612] jobdir: /tmp/openmpi-sessions-daniel.macario@mail.mcgill.ca@tr5130gu-10.ece.mcgill.ca_0/22357/1
[tr5130gu-10.ece.mcgill.ca:03612] top: openmpi-sessions-daniel.macario@mail.mcgill.ca@tr5130gu-10.ece.mcgill.ca_0
[tr5130gu-10.ece.mcgill.ca:03612] tmp: /tmp
[tr5130gu-10.ece.mcgill.ca:03612] [[22357,1],2] node[0].name tr5130gu-13 daemon 0
[tr5130gu-10.ece.mcgill.ca:03612] [[22357,1],2] node[1].name ECE12 daemon 1
[tr5130gu-10.ece.mcgill.ca:03612] [[22357,1],2] node[2].name ECE13 daemon 2
[tr5130gu-10.ece.mcgill.ca:03612] [[22357,1],2] node[3].name ECE10 daemon 3
[tr5130gu-10.ece.mcgill.ca:03613] procdir: /tmp/openmpi-sessions-daniel.macario@mail.mcgill.ca@tr5130gu-10.ece.mcgill.ca_0/22357/1/5
[tr5130gu-10.ece.mcgill.ca:03613] jobdir: /tmp/openmpi-sessions-daniel.macario@mail.mcgill.ca@tr5130gu-10.ece.mcgill.ca_0/22357/1
[tr5130gu-10.ece.mcgill.ca:03613] top: openmpi-sessions-daniel.macario@mail.mcgill.ca@tr5130gu-10.ece.mcgill.ca_0
[tr5130gu-10.ece.mcgill.ca:03613] tmp: /tmp
[tr5130gu-10.ece.mcgill.ca:03613] [[22357,1],5] node[0].name tr5130gu-13 daemon 0
[tr5130gu-10.ece.mcgill.ca:03613] [[22357,1],5] node[1].name ECE12 daemon 1
[tr5130gu-10.ece.mcgill.ca:03613] [[22357,1],5] node[2].name ECE13 daemon 2

```
[tr5130gu-10.ece.mcgill.ca:03613] [[22357,1],5] node[3].name ECE10 daemon 3
>>>read matrix successfully!
Number of rows: 2000
Number of cols: 2001
Number of processes 8
Rows per process: 250
Rows for last process 250
Best profit 5.960190 given by Treshold 0.600000
Algorithm times: 33980815
[tr5130gu-13.ece.mcgill.ca:15675] sess_dir_finalize: proc session dir not empty - leaving
[tr5130gu-13.ece.mcgill.ca:15675] sess_dir_finalize: proc session dir not empty - leaving
[tr5130gu-13.ece.mcgill.ca:15676] sess_dir_finalize: proc session dir not empty - leaving
[tr5130gu-13.ece.mcgill.ca:15678] sess_dir_finalize: proc session dir not empty - leaving
[tr5130gu-13.ece.mcgill.ca:15677] sess_dir_finalize: proc session dir not empty - leaving
[tr5130gu-13.ece.mcgill.ca:15675] sess_dir_finalize: proc session dir not empty - leaving
[tr5130gu-13.ece.mcgill.ca:15675] sess_dir_finalize: job session dir not empty - leaving
[tr5130gu-12.ece.mcgill.ca:02394] sess_dir_finalize: proc session dir not empty - leaving
[tr5130gu-10.ece.mcgill.ca:03613] sess_dir_finalize: proc session dir not empty - leaving
[tr5130gu-12.ece.mcgill.ca:02393] sess_dir_finalize: proc session dir not empty - leaving
[tr5130gu-10.ece.mcgill.ca:03611] sess_dir_finalize: proc session dir not empty - leaving
[tr5130gu-12.ece.mcgill.ca:02392] sess_dir_finalize: proc session dir not empty - leaving
[tr5130gu-12.ece.mcgill.ca:02392] sess_dir_finalize: proc session dir not empty - leaving
[tr5130gu-12.ece.mcgill.ca:02395] sess_dir_finalize: proc session dir not empty - leaving
[tr5130gu-12.ece.mcgill.ca:02392] sess_dir_finalize: proc session dir not empty - leaving
[tr5130gu-12.ece.mcgill.ca:02392] sess_dir_finalize: job session dir not empty - leaving
[tr5130gu-10.ece.mcgill.ca:03612] sess_dir_finalize: proc session dir not empty - leaving
[tr5130gu-13.ece.mcgill.ca:15668] [[22357,0],0] Releasing job data for [22357,1]
[tr5130gu-10.ece.mcgill.ca:03611] sess_dir_finalize: proc session dir not empty - leaving
[tr5130gu-10.ece.mcgill.ca:03611] sess_dir_finalize: job session dir not empty - leaving
[tr5130gu-13.ece.mcgill.ca:15675] sess_dir_finalize: job session dir not empty - leaving
[tr5130gu-10.ece.mcgill.ca:03611] sess_dir_finalize: job session dir not empty - leaving
[tr5130gu-13.ece.mcgill.ca:15668] sess_dir_finalize: job session dir not empty - leaving
[tr5130gu-13.ece.mcgill.ca:15668] [[22357,0],0] Releasing job data for [22357,0]
[tr5130gu-13.ece.mcgill.ca:15668] sess_dir_finalize: proc session dir not empty - leaving
orterun: exiting with status 0
[tr5130gu-12.ece.mcgill.ca:02392] sess_dir_finalize: job session dir not empty - leaving
[daniel.macario@mail.mcgill.ca@tr5130gu-13 ~/Documents/FALL_2015/ECSE_420/
assignmentSolutions/A4_2]$
```

Below is the config file that I placed in the .ssh folder of my home directory:

```
Host ECE15
	HostName TR5130GU-15.ECE.McGill.CA
	User daniel.macario@mail.mcgill.ca

Host ECE13
	HostName TR5130GU-13.ECE.McGill.CA
	User daniel.macario@mail.mcgill.ca

Host ECE12
	HostName TR5130GU-12.ECE.McGill.CA
	User daniel.macario@mail.mcgill.ca
```

```
Host ECE11
	HostName TR5130GU-11.ECE.McGill.CA
	User daniel.macario@mail.mcgill.ca

Host SOCS
	Hostname mimi.cs.mcgill.ca
	User dmacar2
```