

CS 290B - Homework 1: Experimental Results

We shall see two kinds of experiments that were done to compute the total time to get the tasks executed on the server and analyse each of them separately. Each task was run 5 times and the average response time was calculated.

1. Client and Server on the same machine but on two JVMs:

In the first experiment, the client and server were made to run on the same machine but on two different address spaces.

Task	Trial 1 (ms)	Trial 2 (ms)	Trial 3 (ms)	Trial 4 (ms)	Trial 5 (ms)	Average Elapsed Time (ms)
MandelbrotSet Task	81	10	9	13	9	24
EuclideanTSP Task	1805	1797	1821	1840	1882	1829

The MandelbrotSet task takes very little time when compared to the Euclidean TSP task. That is due to the fact that the TSP task does a lot more computations to get the minimal tour. Since, there are 10 cities in the given TSP, there are a total of $10!$ permutations and the distance is computed for each of the permutations and the minimum distance is returned. But in the case of MandelbrotSet task, there are only a maximum of little more than 256×256 iterations (depends on whether the chosen point inside the complex region belongs to the Mandelbrot set or not), where 256 refers to the number of small squares in the region.

There's no network latency involved in this particular experiment. Another observation from the table is that "Trial 1" for MandelbrotSet task takes a lot more time compared to the other trials. This might be due to the reason that the server loads the implementation class for the task for the first time from the client's codebase when it's not found in its local codebase. But, for the EuclideanTSP task, there's not much difference between the first trial and the rest of the trials. This is due to the fact that the EuclideanTSP implementation class would have already been loaded when the MandelbrotSet task was loaded as they are part of the same jar file that was downloaded when MandelbrotSet task was executed.

2. Client on my local machine and Server on a CSIL machine

Task	Trial 1 (ms)	Trial 2 (ms)	Trial 3 (ms)	Trial 4 (ms)	Trial 5 (ms)	Average Elapsed Time (ms)

MandelbrotSet Task	206	104	106	103	106	125
EuclideanTSP Task	1913	1887	1906	1909	1906	1904

In this case, we can see that the response times are longer than the previous setup. This can be attributed to the network latency between the two remote machines and it's also due to the additional work involved in serializing and deserializing the data on and off the wire.