Yahya Alhinai

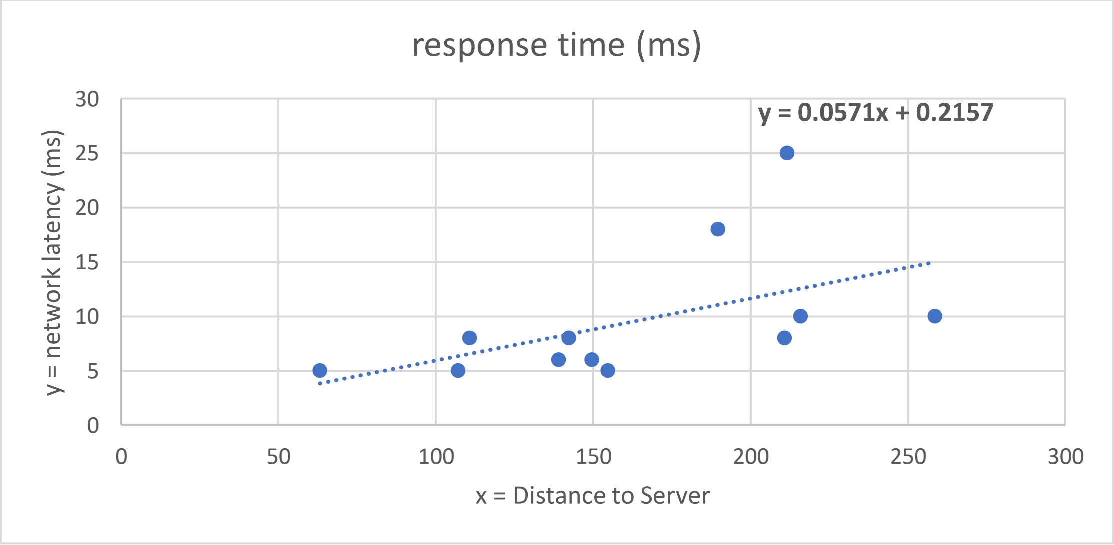
EE5371

November 18th, 2019

Problem 1:

Plot your measured values and your regression model on the same figure:

Driving Distance (Km)	response time (ms)
63.15	5
106.96	5
110.69	8
138.92	6
149.53	6
142.09	8
154.65	5
189.62	18
210.74	8
211.62	25
215.86	10
258.59	10



Prediction of the system response to distance is described by the following equation:

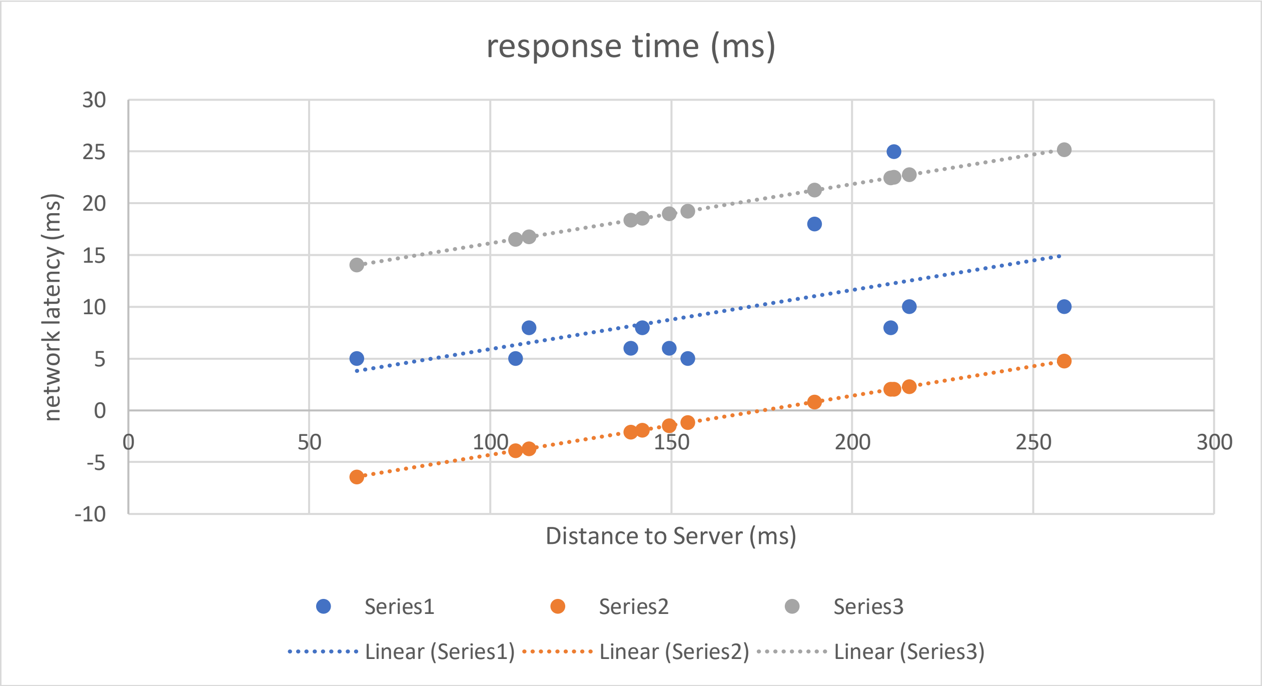
$$network\ latency = 0.0571 * Distance + 0.2157\ (\mathrm{ms})$$

The confidence intervals is chosen to be 90% for the model that determine the range by the equation below:

$$latency\ (upper, lower) = pred.latency \pm t_{[95\%,10]} * S * \sqrt{1 + \frac{1}{n} + \frac{(distance - mean\ distance)^2}{S_{xx}}}$$

The minimum network latency of the system, with 90% confidence interval is shown in the table and graph below:

response time (ms)	driving distance (Km)	Predicted response time (ms)	predicted lower bound (ms)	predicted upper bound (ms)
5	63.15	3.821565	**-6.39425249**	14.03738249
5	106.96	6.323116	**-3.89270149**	16.53893349
8	110.69	6.536099	**-3.67725219**	16.74945019
6	138.92	8.148032	**-2.06668943**	18.36275343
6	149.53	8.753863	**-1.46085843**	18.96858443
8	142.09	8.329039	**-1.88431219**	18.54239019
5	154.65	9.046215	**-1.16960249**	19.26203249
18	189.62	11.043002	**0.820062988**	21.26594101
8	210.74	12.248954	**2.035602814**	22.46230519
25	211.62	12.299202	**2.053288811**	22.54511519
10	215.86	12.541306	**2.328228885**	22.75438312
10	258.59	14.981189	**4.768111885**	25.19426612



Problem 2:

The website (http://speedtest.tele2.net/) That I used to test the time downloading as a function of distance and size. The severs in those 7 locations was used [Croatia, Zagreb - Germany, Frankfurt - Latvia, Riga - Lithuania, Vilnius - Netherlands, Amsterdam - Sweden, Gothenburg - Sweden, Stockholm]. For each location, size of 1MB, 10MB, 100MB, and 1000MB were conducted and the results is shown in the following table:

time (ms)	Size (MB)	Distance (KM)	predicted time (ms)
0.7	1	6684.45	**0.529215**
0.7	1	7044.12	**1.063053**
0.7	1	7766.93	**2.13588**
0.8	1	6681.87	**0.525386**
0.8	1	7514.55	**1.761286**
0.8	1	6842.22	**0.763384**
1.1	1	7280.36	**1.413691**
1.3	10	6842.22	**1.187115**
1.3	10	6681.87	**0.949117**
1.3	10	6684.45	**0.952946**
1.3	10	7766.93	**2.559611**
1.4	10	7044.12	**1.486784**
1.4	10	7514.55	**2.185017**
1.7	10	7280.36	**1.837422**
4.9	100	7514.55	**6.422328**
5.1	100	6681.87	**5.186427**
6.3	100	7280.36	**6.074732**
6.7	100	7044.12	**5.724094**
7	100	6842.22	**5.424426**
7.4	100	6684.45	**5.190256**
7.9	100	7766.93	**6.796921**
40	1000	6842.22	**47.79753**
45	1000	6681.87	**47.55953**
46	1000	7514.55	**48.79543**
47	1000	7044.12	**48.0972**
51	1000	6684.45	**47.56336**
53	1000	7766.93	**49.17002**
55	1000	7280.36	**48.44784**

The coefficients for the model with 90% confidence intervals:

$$Intercept = [-24.537, 5.658]$$

$$Size = [0.0451, 0.0490]$$

$$Distance = [-0.000632, 0.00360]$$

The regression model from the data obtained for time as a function of server distance and file size is represented by:

$$Time = -9.439 + \ 0.0470 * Size + 0.00148 * Distance$$

The predicted time for file downloading is show in the 4th column of the table above.

Problem 3:

1. The size of one file should be within the range of files sizes measured was chosen to be 10MB file size and the server (Germany, Frankfurt) with 7044.12 Km distance from Minneapolis, MN:

$$Time = -9.439 + \ 0.0470 * Size + 0.00148 * Distance$$

trials	Size	distance	predicted time	measured time
1	10 MB	7044.12 Km	1.486784 s	1.4 s
2	10 MB	7044.12 Km	1.486784 s	1.3 s
3	10 MB	7044.12 Km	1.486784 s	1.5 s
4	10 MB	7044.12 Km	1.486784 s	1.4 s
5	10 MB	7044.12 Km	1.486784 s	1.5 s
6	10 MB	7044.12 Km	1.486784 s	1.6 s
7	10 MB	7044.12 Km	1.486784 s	1.5 s
8	10 MB	7044.12 Km	1.486784 s	1.4 s

From the measured times, produce the following statistical information:

$$mean = \ 1.45 \ s$$

$$confidence \ interval \ of \ 90\% = [1.387, 1.512]$$

$$predicted \ time \ to \ transfer \ 10MB \ \ = 1.486 \text{ s}$$

2. The size of the file that is larger than the largest file measured is 10GB:

trials	Size	distance	predicted time	measured time
1	10,000 MB	7044.12 Km	471.8282342 s	**446.4 s**
2	10,000 MB	7044.12 Km	471.8282342 s	**501.0 s**
3	10,000 MB	7044.12 Km	471.8282342 s	**486.6 s**
4	10,000 MB	7044.12 Km	471.8282342 s	**454.2 s**
5	10,000 MB	7044.12 Km	471.8282342 s	**444.6 s**
6	10,000 MB	7044.12 Km	471.8282342 s	**488.4 s**
7	10,000 MB	7044.12 Km	471.8282342 s	**433.8 s**
8	10,000 MB	7044.12 Km	471.8282342 s	**486.6 s**

From the measured times, produce the following statistical information:

$$mean = \ 467.7 \ s$$

$$confidence \ interval \ of \ 90\% = [450.58, 484.81]$$

$$predicted \ time \ to \ transfer \ 10GB \ \ = 471.82 \text{ s}$$

The prediction time to transfer 10 MB and 10GB are within the range of 90% confidant interval which mean the prediction is actually good.

Problem 4:

1. Transferring 100 PB of data:

The size of the file transfer is 100PB file and the server (Germany, Frankfurt) with 7044.12 Km distance from Minneapolis, MN. To estimate the time required to transfer 100PB size of data with my internet speed connection:

$$Time = -9.439 + \ 0.0470 * Size + 0.00148 * Distance$$

The time estimated is:

$$estimated\ time = -9.439 + \ 0.0470 * 10^{11} + 0.00148 * 7044.12$$

$$estimated\ time = 4.71 * 10^{9}\ s = \boldsymbol{1307812\ hours}$$

Assuming the internet connection never shut dawn and we have enough space to store the data downloading. The estimated time to transfer 100PB of data from the server (Germany, Frankfurt) is 1307712 hours of downloading.

To estimate the speed of a semi-trailer truck that matches our estimated time to transfer 100PB:

$$\text{semi\_trailer truck}\ speed = \frac{distance}{time} = \frac{7044.12}{1307812} = 0.005386\ \frac{\text{km}}{\text{h}} = 5.3861\ \frac{m}{h}$$

Assuming the truck is moving in a straight line from Minneapolis to Frankfurt, the speed of the truck that would require the same amount of time to transfer 100PB as my internet connection is **5.386** meter per hour which it would reach its destination in 1307812 hours. This is setup to match the same amount of time that required downloading 100PB at my internet speed.

It's for sure better to transfer 100PB of data physically from Minneapolis to Frankfurt via shipping container pulled by a semi-trailer truck rather than downloading the file from the server. This is because a truck can have faster speed.

2. Various data transferring to estimate the break-even point:

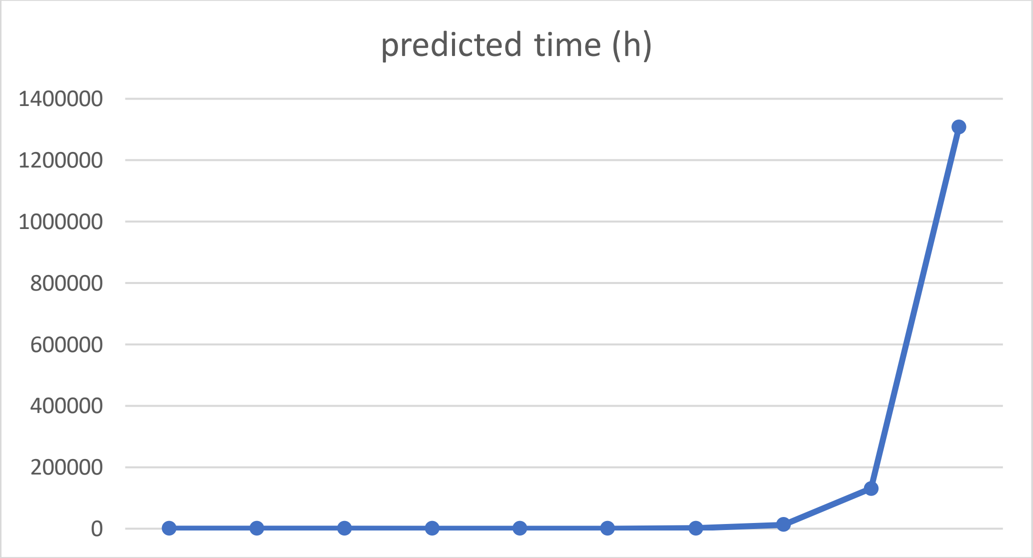
This table test for various file size for one server (Germany, Frankfurt):

Data transferring (MB)	Distance (Km)	predicted time (s)	predicted time (h)	Speed of truck required (Km)
100	7044.12	5.724094	0.00159	4430191
1000	7044.12	48.0972	0.01336	527241.4
10000	7044.12	471.8282	0.131063	53745.9
100000	7044.12	4709.139	1.308094	5385.026
1000000	7044.12	47082.24	13.0784	538.6071
10000000	7044.12	470813.3	130.7815	53.86176
100000000	7044.12	4708124	1307.812	5.386187

1000000000	7044.12	47081227	13078.12	0.538619
10000000000	7044.12	4.71E+08	130781.2	0.053862
100000000000	7044.12	4.71E+09	1307812	0.005386

As table above shown, the break-even point for data to be transfer physically from Minneapolis to the location of the sever rather than downloading the file from the server is found to be 100TB or higher (Highlighted in gray). At 100TB, the time required to download the file would be around 1307 hours which means it's going to be better to transfer it physically time-wise as well as financial-wise.

We can observe the exponential increase in the time required to download the file. The same trend is seen in different server distances. The following shows the behavior trend of the time required to transfer data as a function with size of the file increasing by a factor of 10:



Problem 5:

This paper introduces three parallel performance laws that provide operational analysis to analyze a client-server environment and estimates of queueing times as well as the performance implications of capacity expansion and reduction. Those three concepts presented: the occupancy law, the capacity adjustment law, and the capacity expansion law.

As the three laws were modeled for a real parallel network server, it only required 5 assumptions to be true in order for the three laws presented in the paper be effetely describing a real parallel network server. The techniques presented in Chapter 11 required more assumptions to be made and it only account to describe the trend behavior of the parallel system in an ideal situation gives rigid measurement instead of the actual parallel system behavior which. Therefore, the techniques presented in this paper is more accurate in describing real parallel system. Finally, the paper presents an applicable to the real word measurement of parallel network server

Problem 6:

What is the problem being studied?

The scope of the paper is meant to identify what DMBS parameters are most effecting the improving performance and the methodology to find those DMBS parameters and tune them.

Is this an important problem? Why or why not?

As the more and more data is being produced and process. The problem that this paper is trying to address is very crucial as the size of data sets are growing exponentially with time. Meaning the mining of ever-increasing datasets is getting harder and harder.

What are the main results?

The main issue this paper tackling is finding a way that is systematic and consistence in ranking "the configuration parameters based on the impact on DBMS performance for the entire query workload" as well as "selecting a compressed query workload based on the similarities of performance bottleneck parameters that preserves the fidelity of the original workload." (Debnath et al)

What method is used to produce these results?

The method applied is a full factorial design which quantify the impact of all factors and its interactions.

What are the main assumptions? How realistic are these assumptions? How sensitive are the results to the assumptions?

Since full factorial design measured for all possible input combinations which requires an exponential number of experiments, two assumptions were made to reduce the number of experiments.

First assumption, for each configuration parameter consider only two values minimum and maximum which gives the maximum range of output responses for each input. Second assumption, only few main factors largely dominate system response as well as the some lower-order interactions.

Those two assumptions are very applicable as it was demonstrated in the paper that large performance changes from a few dominating factors. Those DMBS interactions are sensitive to any parameter changes and can affect the performance greatly.

What did you learn from this paper?

This paper has taught me a systematic and consistence method that effectively pick out the most effecting inputs in a possibly large set of data by using the assumptions presented in this paper. It works great to a wild variety of applications.