MANAGEMENT AND CONTENT DELIVERY FOR SMART NETWORKS

Algorithms and Modelling

A.Y. 2015/2016

LABORATORY 1

PERFORMANCE OF SINGLE QUEUES

BARUSSO FEDERICO

CERUTTI MARCO

ORLANDO MATTEO

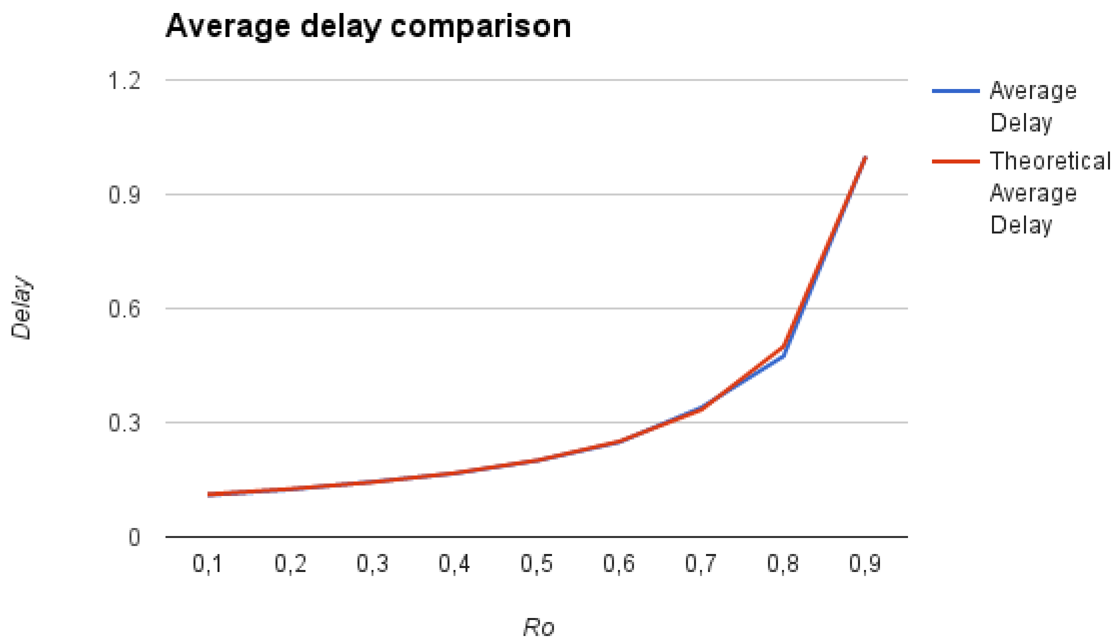
The goal of this laboratory is to practice with queue simulators. Starting from the simulator of the M/M/1 queue, the objective is to get familiar with main elements of the simulator and then, by plotting the results, to evaluate the performance of the queue through simulations.

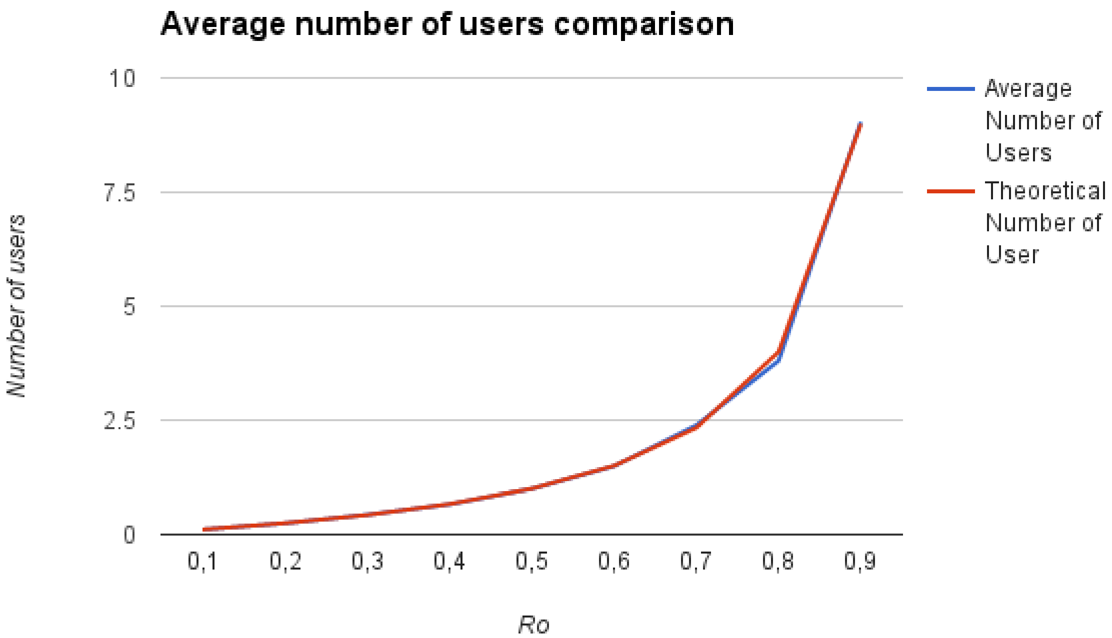
M/M/1 queue simulation

In this first step the simulation is run with a fixed service rate of $100\ customers/s$ and varying the arrival rate from 10 to $90\ customers/s$. The simulation time is $1000s$. No changes in the code.

ρ	number of services	average delay	theoretical average delay	average number of users	theoretical number of user	probability server is idle
0,1	10030	0,110250	0,111111	0,110692	0,111111	0,9
0,2	19960	0,123732	0,125000	0,247002	0,250000	0,8
0,3	30116	0,143576	0,142857	0,432393	0,428571	0,7
0,4	39879	0,165825	0,166667	0,661364	0,666667	0,6
0,5	50044	0,199336	0,200000	0,997698	1,000000	0,5
0,6	60119	0,248110	0,250000	1,491592	1,500000	0,4
0,7	70391	0,339329	0,333333	2,388774	2,333333	0,3
0,8	80139	0,474599	0,500000	3,803814	4,000000	0,2
0,9	90398	1,000719	1,000000	9,046151	9,000000	0,1

The following plots represent the comparison between the value of delay and number of users obtained with the simulation and the theoretical ones.





In this simple case the results are very similar to the theoretic ones. As predicted, the number of users in the system increases as the arrival rate increase and as a consequence also the average delay increases.

M/M/1 queue simulation: different simulation duration

In this other simulation, the simulation time is the first time $1000s$ and the second time $10000s$.

λ	μ	Time	number of services	average delay	theoretical average delay	average number of users	theoretical number of user
5	8	1000	4940	0,325827	0,333333	1,611180	1,666667
5	8	10000	49754	0,322042	0,333333	1,602284	1,666667

Multiplying by 10 the simulation time, also the number of services is more or less 10 times higher. The other measured values remain almost the same. In this case the simulation time doesn't affect the simulation results.

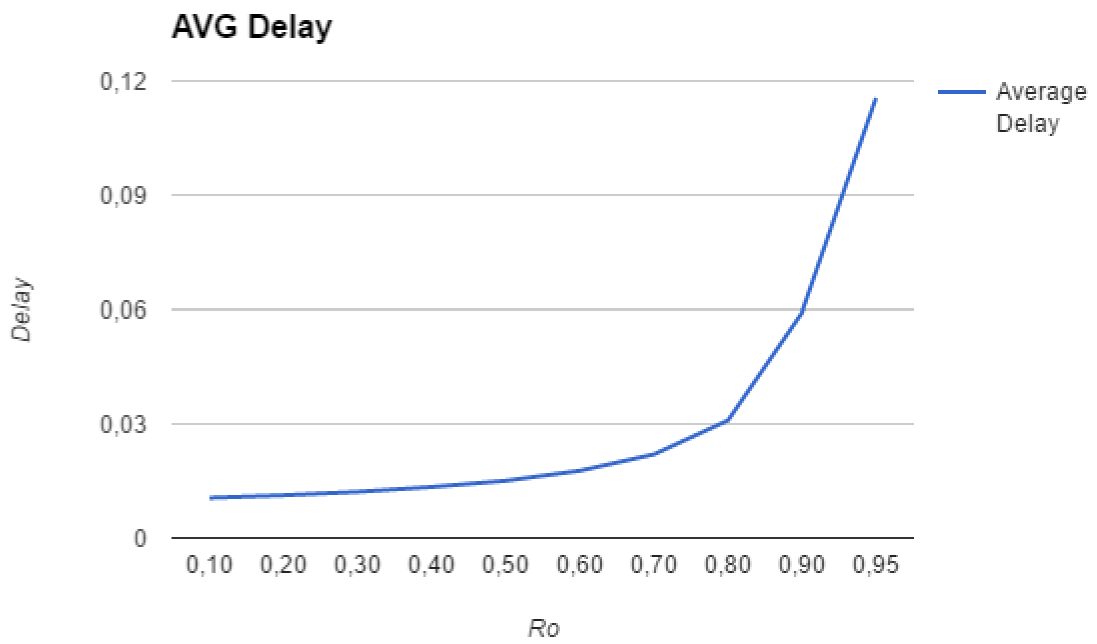
M/G/1 queue simulation

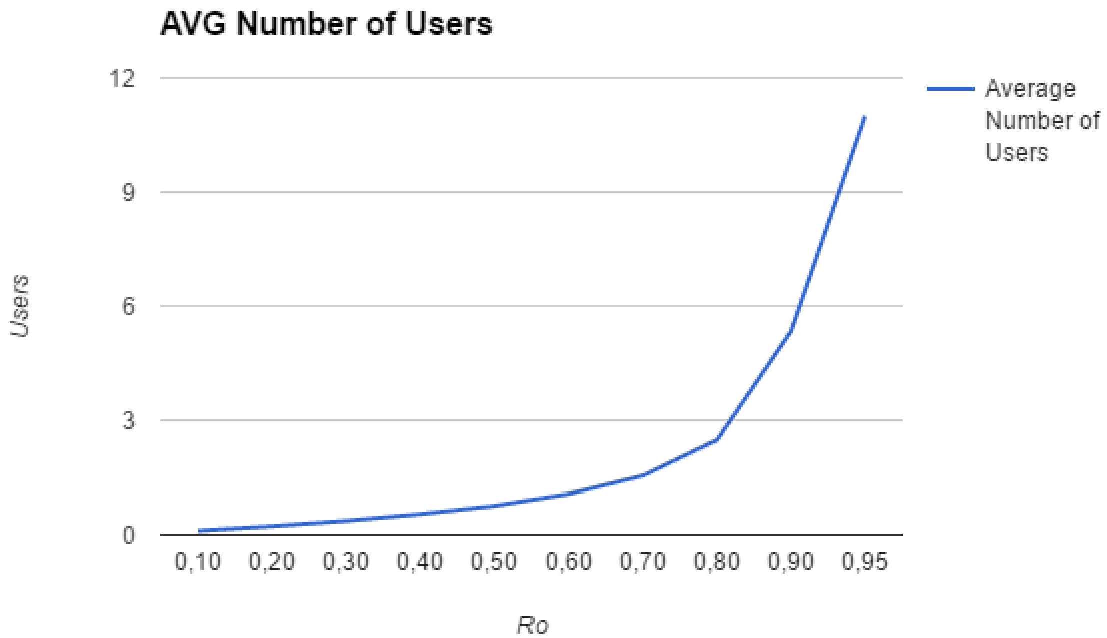
In this simulation the exponential distributed service time typical of M/M/1 queues is substituted by one with a different distribution. The chosen one is the constant distribution, because makes calculations simpler and also because it models in a good way the behaviour of today's routers.

As before the service rate is fixed, while the ratio between the arrival rate and the service rate varies from $0{,}1$ to $0{,}95$. The $0{,}95$ value is included because there is more interest in analyzing a queue behavior in stress conditions, because are the worst work conditions. The simulation time is fixed to $1000s$.

ρ	Number of Services	Average Delay	Theoretical Average Delay	Average Number of Users	Theoretical Number of User
0,1	10035	0,010555	0,010556	0,106025	0,105556
0,2	20128	0,011245	0,011250	0,226601	0,225000
0,3	30197	0,012146	0,012143	0,366877	0,364286
0,4	40310	0,013417	0,013333	0,540829	0,533333
0,5	50165	0,015054	0,015000	0,755207	0,750000
0,6	60312	0,017699	0,017500	1,067481	1,050000
0,7	70149	0,022011	0,021667	1,544012	1,516667
0,8	80160	0,030940	0,030000	2,481128	2,400000
0,9	90144	0,059234	0,055000	5,340560	4,950000
0,95	95168	0,115527	0,105000	10,99504	9,975000

The following plot represent the behavior of the average delay and of the average number of users in this M/G/1 simulation.





Having these data is possible to verify the **Pollaczek-Khintchin** formula:

$$\boxed{E[N] = \rho + \rho^2 \frac{(1 + C_S^2)}{2(1 - \rho)}}$$

$$\boxed{E[T] = E[S] + \rho E[S] \frac{(1 + C_S^2)}{2(1 - \rho)}}$$

Where $E[N]$ is the average number of customers in the system, ρ is the ratio between the arrival and the service rate, C_S^2 is defined as $E[S^2] - E^2[S] / E^2[S]$ and is constant and equal to zero in this queue (M/D/1 type), and $E[T]$ is the average delay.

In this simulation the service time for the single customer is set to $E[s] = 0.01$.

In the following table are shown the results obtained by the PK formula and the difference with the measured and the theoretical ones.

E[N]_PK	E[N] PK-measured	E[N] PK-theoretical	E[T]_PK	E[T] PK-measured	E[T] PK-theoretical
0.105556	0.000469	0.000000	0.010556	0.0000006	0.000000
0.225000	0.001601	0.000000	0.011250	0.0000050	0.000000
0.364286	0.002591	0.000000	0.012143	0.0000031	0.000000
0.533333	0.007496	0.000000	0.013333	0.0000837	0.000000
0.750000	0.005207	0.000000	0.015000	0.0000540	0.000000
1.050000	0.017481	0.000000	0.017500	0.0001990	0.000000
1.516667	0.027345	0.000000	0.021667	0.0003443	0.000000
2.400000	0.081128	0.000000	0.030000	0.0009400	0.000000
4.950000	0.390560	0.000000	0.055000	0.0042340	0.000000
9.975000	1.020040	0.000000	0.105000	0.0105270	0.000000

Is possible to see that the differences between the values obtained by the PK formula and the theoretical ones is always zero, as expected, so the PK formula is verified.

Besides is possible to see that the difference between the measured values and the ones obtained by the PK formula is always really little, so that is a proof of the goodness of the simulator.

M/G/k/B queue simulation

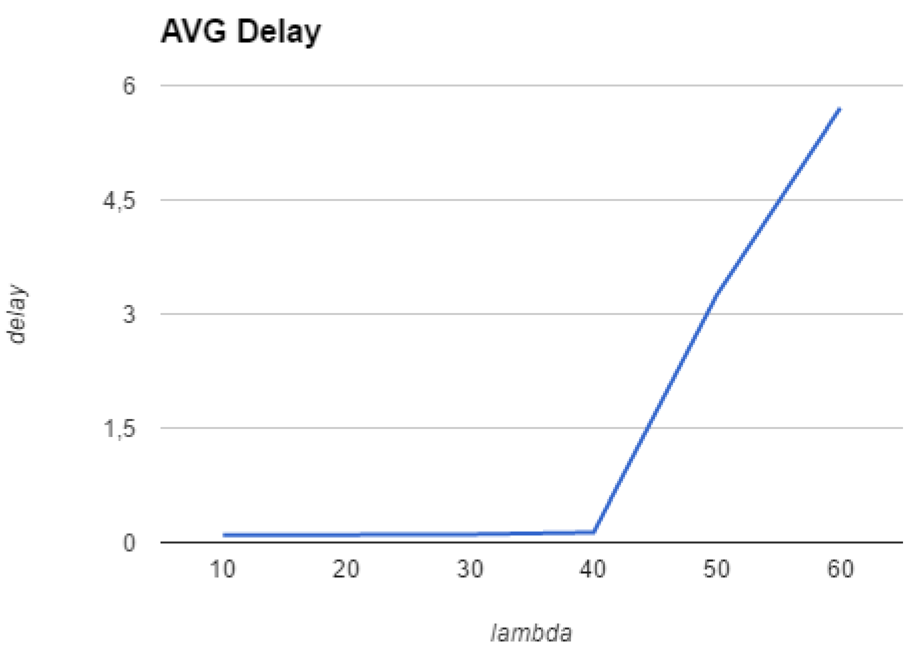
For the implementation of the M/G/k/B queue the original code has been modified inserting a limitation on the length of the queue and verifying each time a customer arrives, if the number of user in the queue don't overcome its dimension. Then the *in_service Record** variable was substituted with a vector of k elements.
All statistics are referred to the complete system with a limited queue and k servers.

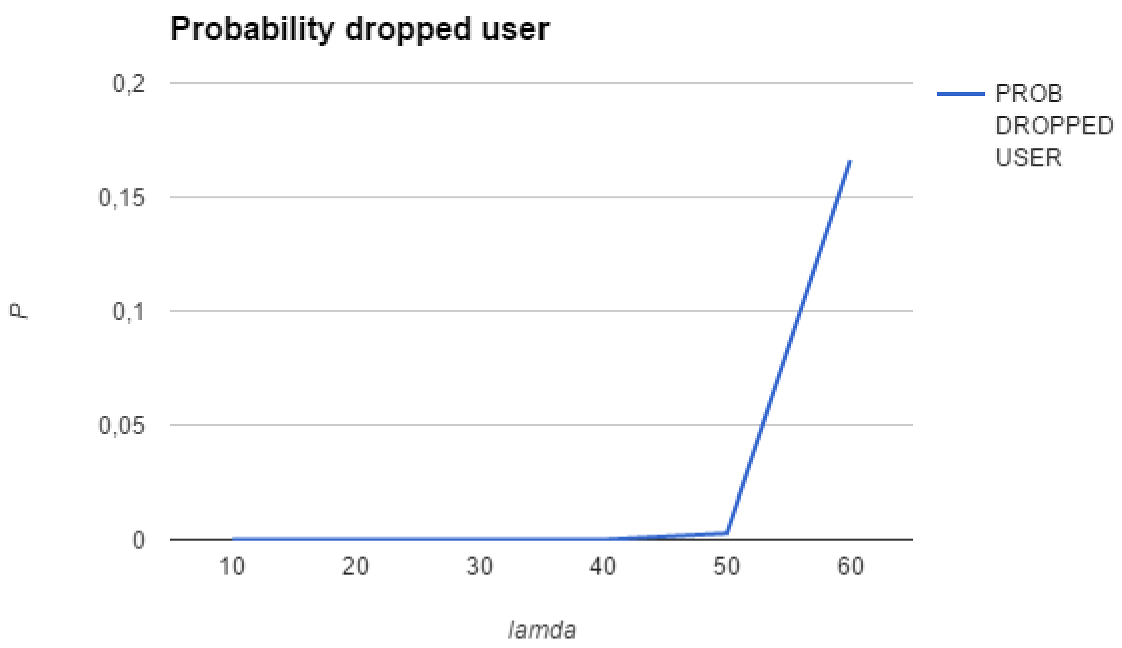
In the following table are shown the statistics about the simulation. The system is one queue with dimension $k = 300$ and 5 servers with a constant service rate of 10 $customers/s$. The simulation time is 1000 unit.

λ	AVERAGE DELAY	AVERAGE # USERS	AVERAGE # BUSY SERVER	# USERS SERVED	# OF LOSSES	PROBABILITY OF USER DROP
10	0.100060	1.004050	2.006776	10035	0	0.000000
20	0.101280	2.038361	2.968950	20128	0	0.000000
30	0.107001	3.231072	3.798397	30197	0	0.000000
40	0.131962	5.319178	4.486976	40307	0	0.000000
50	3.252983	162.461347	4.995722	49907	141	0.002811
60	5.705532	286.108935	4.999801	49995	10026	0.166233

Is possible to see that when λ is lower than $5 * service\_rate$, the system is ergodic and there are no losses. When λ reaches 50 $customers/s$ the system begins to be unstable and most of the time all servers work. Customers begin to be lost. If λ grows again, the load overcomes 1 and the system is totally unstable so a very high number of customers are lost.

In the following plots are represented the average delay and the probability of user drop.





In the next table is shown how many customers are served in the simulation by each servers.

Lambda	SERVER1	SERVER2	SERVER3	SERVER4	SERVER5
10	5033	2988	1415	476	123
20	6788	5491	3993	2541	1316
30	7969	7217	6264	5036	3712
40	8981	8671	8229	7633	6796
50	9993	9989	9984	9978	9968
60	10000	10000	10000	10000	10000

Since the service rate of each server is constant ($10\ customer/s$) and the simulation time is $1000s$, it means that each server can serve a maximum number of 10000 customers during the simulation. So when λ is equal to 60, on average, 60000 customers arrive, but 5 servers cannot serve all of them. In this special case 10000 customers on average are lost.

From this table it is evident what is the customer-server assignment's policy: when a customer arrives it is assigned to the first free server starting from SERVER1. In fact, is possible to see, checking also the average number of busy servers from the previous table, that when λ is low, so the load is low, the last servers do not work for a large amount of time. Only the firsts servers are a great number of customers to serve. Obviously this difference is much little increasing λ. Already when λ is equal to $40\ customers/s$, most of the time all servers work. Indeed, when the load is critical, on average all servers are used.

To control that the system works well, k is set to 1, and very long queue is defined, so is possible compare the results with the previous M/G/1 queue.

ρ	AVERAGE DELAY M/G/1	AVERAGE DELAY M/G/k/B	AVERAGE # USER M/G/1	AVERAGE # USER M/G/k/b
0,1	0,010555	0,0104624	0,106025	0,101213
0,2	0,011245	0,0111868	0,226601	0,223277
0,3	0,012146	0,0120171	0,366877	0,357461
0,4	0,013417	0,0134795	0,540829	0,541390
0,5	0,015054	0,0149647	0,755207	0,748136
0,6	0,017699	0,0174834	1,067481	1,050895
0,7	0,022011	0,0216782	1,544012	1,513503
0,8	0,03094	0,0309240	2,481128	2,475194
0,9	0,059234	0,0581227	5,34056	5,228947

The obtained results are very similar, depending also on the chosen seed.