Computer Engineering 4DK4
Lab 3
Call Blocking in Circuit Switched Networks

This lab first investigates call blocking in a circuit switched network. Call arrivals occur to a cellular basestation containing N communication channels. If a channel is available when a call arrives, it is assigned to the call for its duration, otherwise the call is blocked. A computer simulation is used to assess the blocking behaviour of the system with Poisson process call arrivals and exponentially distributed call holding times. The simulation results are compared to theory using the Erlang B formula. The case is then considered where incoming calls are not blocked but permitted to queue up until service is available. This system is simulated and compared to theory using the Erlang C formula. Finally, code is written that simulates a company that operates a fleet of N taxis that service incoming call requests. Results are obtained that assess the behaviour of the system.

1 Preparation

1. An electronic copy of the simulation code must first be obtained.
2. You must compile and run the simulation. If you are not familiar with C and a C compiler make sure you see me or one of the TAs as soon as possible.

2 Experiments

1. As in the other labs, first familiarize yourself with the code and with running the simulation. Make sure that you understand how this simulation works. As before, the parameters for the simulation runs are in `simparameters.h`. *In all the experiments make sure you include runs using your McMaster student ID number as the random number generator seed.*
2. Using the provided simulation code, generate a set of curves that show the tradeoffs between blocking probability, offered load (in Erlangs) and the number of channels. (Curves of this kind were presented earlier in class. See the "Circuit Switching Performance" lecture overheads.)

   Compare your results to that obtained using the Erlang B formula. According to Erlang B, the probability that a call is blocked is given by

$$P_B = \frac{A^N/N!}{\sum_{i=0}^{N} A^i/i!}$$

where A is the offered load (in Erlangs) and is given by

$$A = \lambda h$$

where λ is the call arrival rate (in calls per unit time, e.g., calls/minute), h is the average call holding time (in the same time units, e.g., minutes), and N is the total number of channels. Write a program to perform this computation (Feel free to use any programming language you want, e.g., Matlab, C, etc.) Include a listing of your program in your writeup. Check the results of your program using one of many online Erlang B calculators. Just google search for "Erlang B calculator". Include the URL of the calculator that you used.

3. Cellular networks are often designed so that under busy traffic conditions, the highest acceptable probability of blocking is about 1% (i.e., the probability of rejecting a cell-phone call is 1% because the cellular network has no available channels.) Generate a graph of maximum offered loading (in Erlangs) versus the number of cellular channels needed to achieve this blocking probability performance.

4. Consider a company that operates a call centre where customers phone in to make enquiries. We can reasonably assume that customer enquiries arrive according to a Poisson process and that each enquiry takes a random exponentially distributed time (with a mean of h seconds) to be served. The call centre has a staff of N operators who respond to the incoming calls. Unlike the Erlang B case, if all N operators are busy, the caller is placed in a FIFO queue listening to horrible music. Customers that wait on this queue however, will eventually hang up and leave the system if they wait too long. From the time that a customer is placed in the queue, the time until they hang up is exponentially distributed with a mean of w seconds.

   Modify the simulation to model the above situation. Generate graphs that show the probability of not being served (because of a hang-up) and the mean time that customers have to wait on the queue versus offered loading for different values of w and N.

5. In a cellular system such as that considered in Part 2, assume that callers never hang up when all channels are found busy, but instead, they wait until their call can be served. Make some changes to the supplied code so that you can simulate this case.

   In this case, the probability of an arriving call having to wait for service can be derived as

   $$P_w = \frac{A^N/N!}{A^N/N! + (1-\rho)\sum_{i=0}^{N-1} A^i/i!}. \tag{1}$$

   where $\rho = A/N$. Note that since there is no blocking in this system, we must have $\rho < 1$ for stability. Equation 1 is referred to as the Erlang C Formula and is commonly used to design telephone call centres. It can be shown that the average caller waiting time is given by

   $$T_w = \frac{P_w h}{N(1 - A/N)} \tag{2}$$

   and the probability that a call has to wait less than t seconds is given by

   $$W(t) = 1 - P_w e^{-(N-A)t/h}. \tag{3}$$

   Select a set of system parameters and compare output from your simulation to that obtained from $W(t)$.

3 Writeup

Submit a writeup for the lab.

1. Each group (of 2 maximum) should submit a single writeup.
2. Include in your writeup a description of everything that you did including all data and the random number generator seeds that were used to obtain the graphs.
3. Include with your writeup the plots and a listing of any of the code that you changed.
4. Unless otherwise indicated, your performance data and curves should cover the system operations from low traffic load to high traffic load.
5. No piecewise linear plots or overfitted curves are allowed.