Homework 3

Due Date for Problems: Feb 13, 2017
Due Date for Lab: Feb 20, 2017

Problem 1: A wireless BS is serving download requests to different users. Because of their different ranges from the BS, users are split into 'far' and 'near' users:

- Near user requests arrive as a Poisson process with rate $\lambda_n = 0.25$ files per second, and far user requests arrive as a Possion process with rate $\lambda_f = 0.1$ files per second.
- Near users request files with mean size $10Mbit$ and coefficient of variation 2. Far users request files with mean size $5Mbit$ and coefficient of variation 1.
- The BS operates as a FCFS queueing system.
- Whenever a file from a near user is at the head of the queue it will be downloaded at a speed of 10Mbps (better channel) while a file for a far user at 1Mbps (bad channel).

Questions:

1. What is the utilization ρ of this system?
2. What is the mean download time (queueing + transmission) for files to near users? What about for files to near users?
3. What is the mean download time overall?
4. What would be the mean download time (over all files) if the BS operated as a Processor Sharing system instead?

Problem 2: Wiki your way through

The existence of short paths between nodes is a characteristic feature of a number of networks around us. In this exercise, we will look for them in the Wikipedia web graph.

We have listed below ordered pairs of "entities" (objects or people), each having a dedicated web-page on Wikipedia. For each pair, starting from the web-page for the first entity, you have to find a sequence of links (only to other

Wikipedia pages) that will take you to the Wikipedia web-page corresponding to the second. Write down (i) the first path that you found, and (ii) the shortest path that you found. In your submission, specify your paths precisely, i.e., write the sequence of links you used.

- Pareto Distribution — South Korea
- Cheeseburger — Political Philosophy

You get some extra points each if your shortest path is the shortest among all the homework submissions. Note that you are not allowed to edit any Wikipedia pages to "create" your path!

Problem 3: Consider a graph of N nodes ($N \gg 1$) constructed as follows:

- nodes are arranged first in a star topology, with 1 node in the center and $N-1$ "leaf" nodes around, each with one link to the center node.
- each link between *consecutive* leafs is also introduced independently with probability p.

One such sample graph is shown in Figure 1(left)

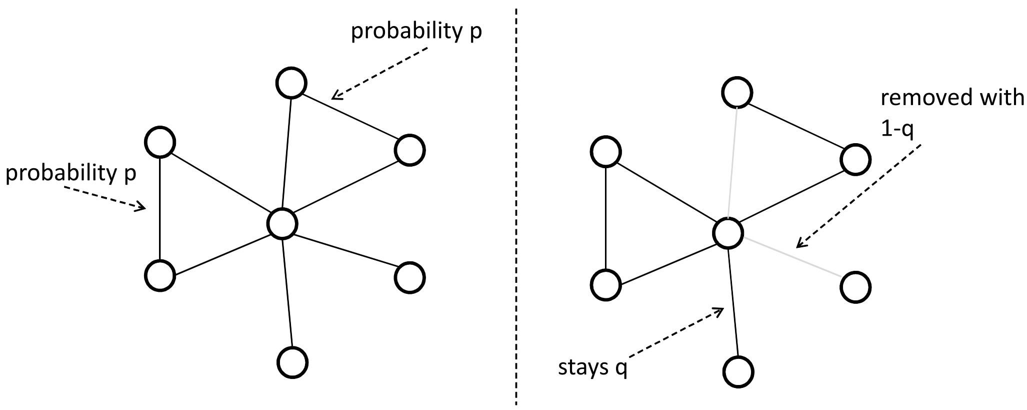

Figure 1:

Answer the following questions:

1. Calculate the degree distribution for this graph.
2. Calculate the average degree.
3. Calculate the clustering coefficient.

4. Assume that we now start removing each star edge with probability $1 - q$ (i.e. the edge between the central node and the chosen leaf node remains there with probability q), as shown in Figure 1(right). Find the threshold value of q, for which there is no more a giant component? (Note: remember that N is large, and the definition of a giant component - and be smart about how you approach the problem).

Problem 4: In most networks, links are not drawn completely at random, but with a tendency to connect nodes which share some common traits. As a simplistic model assume that nodes have different colors, such that there are N red nodes and N blue nodes (assume that N is large). The probability for an edge between nodes of identical color is p and the probability for an edge between nodes of different color is q. Thus for normal networks $p > q$. For $q = 0$ the result is a network of two disjoint clusters each consisting of unicolored nodes.

(a) What is the minimal values of p and q in order to reach global connectivity (with a high probability)?

(b) Is the resulting network small world? If so, for what values of q? Prove or disprove.

(c) You decide to sample this network using a simple random walk (not MCMC). You start your random walk from a red node. How long will it take on average to sample the first blue node?

(d) At what rate does the above random walk converge to its stationary distribution, as a function of p, q, and N.

Queueing Lab: You are required to program in Matlab an M/G/1 queue, apply different scheduling algorithms, and measure the performance differences. This requires a few 10s of lines of Matlab code, using commands that you can quickly learn from any online tutorial. Alternatively, you could also use C or Java to program your queue. The tasks you need to fulfill are the following:

1. Simulate an M/G/1/FCFS queue. Jobs arrive as a Poisson process with rate λ jobs/sec. 98% of jobs have a service requirement of 1sec, while 2% of the jobs have a service requirement of 201sec. Plot the expected mean respone $E[T]$ for this queue as a function of ρ. Choose values of λ so as ut consider a range of ρ from 0.1 to 0.9 (you can pick intermediate values in increments of 0.1). Plot also the theoretical values for each simulation point. Do the simulations agree with the theoretical predictions? (make sure your simulations have converged)

2. Now simulate the same system but implementing Shortest Job First (SJF), non-preemptive, instead of FCFS. Plot again the mean response as a func-

tion of $E[T]$ as a function of ρ. Do that values agree with the theoretical prediction? What do you observe in comparison to FCFS?

3. (Extra Credit) Can you do any better than SJF, *without using a preemptive policy*? You can try anything you want, as long as you don't have preemption and the total service capacity is equal to that of the original system. If so, describe your proposed policy, implement it, and plot $E[T]$ again.

Note: You are required to hand in the plots with comments, and also your code. *Your code has to be run correctly when we try it. You also need to comment each line of your code properly.*