Telecommunication Network

Laurel Yu

February 20, 2006

1 Loss System

For a background review, I started the summer vacation scholorship by looking at loss systems. In a telecommunication context, this means that a new coming call is blocked (or lost) if there is no free circuits for that call.

First of all, I made myself familiar with the notations. In a loss system, let C be the total number of circuits available, n be the number of circuits in use, ν be the arrival rate of new call and μ be the departure rate of an existing call. Then, this stochastic process can be defined by transition rates:

$$\begin{aligned} n \to n+1 \quad & \text{with rate } \nu, \qquad n = 0, 1, \ldots, C-1 \\ n \to n-1 \quad & \text{with rate } n\mu, \qquad n = 0, 1, \ldots, C \end{aligned}$$

If $\tilde{\pi} = (\pi(0), \pi(1), \ldots, \pi(C))$ is the stationary distribution of the process, then it satisfies

$$\sum_{n=0}^{C} \pi(n) = 1$$

and

$$\pi_n = \frac{\lambda_0 \lambda_1 \cdots \lambda_{n-1}}{\mu_1 \cdots \mu_{n-1} \mu_n} \pi_0.$$

Then, it is easy to show that

$$\pi(C) = \frac{\phi^C / C!}{\sum_{n=0}^{C} \phi^n / n!} \tag{1}$$

where

$$\phi = \frac{\nu}{\mu}$$

As $\pi(C)$ is the probability that the process stays in the state that the system is full, it approximates the probability that a call is blocked. Then, equation (1) gives an explicit expression for blocking probability and is also known as Erlang's Formula.

2 Loss Network

A loss network consists K nodes and J links. A route is a subset of $\{1, 2, \ldots, J\}$ and for the convenience of notation, let's assume that there are R subsets, that is, there are R possible number of routes that a call can choose from.

2.1 Fixed Routing

When the routing is fixed, a call is blocked if the at least one link on the route on request is full. Let $\mathbf{n} = (n_1, n_2, \ldots, n_R)^T$ be the number of circuits in use for each route and A_{jr} [1] be an indicator function. That is,

$$A_{jr} = \begin{cases} 1 & \text{if link } j \text{ is on route } r, \\ 0 & \text{otherwise.} \end{cases}$$

Now the statespace of the process can be written as $\{\mathbf{n} : A\mathbf{n} \quad \leq \mathbf{C}\}$, where $\mathbf{C} = (C_1, C_2, \ldots, C_J)^T$ and C_j is the capacity of link j.

The transition rates are:

$$\begin{aligned} \mathbf{n} &\to \mathbf{n} + e_r \quad \text{with rate } \nu_r \\ \mathbf{n} &\to \mathbf{n} - e_r \quad \text{with rate } n_r \end{aligned}$$

where $\pm e_r$ represent an arrival (departure) of a call on route r. (check on what Frank Kelly says about something in product form) From that, the probability that a call is blocked on a particular route r is

$$L_r = 1 - \frac{\displaystyle\prod_{n \in S(C - Ae_r)} \sum_{r=1}^{R} \frac{\nu_r^{n_r}}{n_r!}}{\displaystyle\prod_{n \in S(C)} \sum_{r=1}^{R} \frac{\nu_r^{n_r}}{n_r!}} \tag{2}$$

2.1.1 Erlang Fixed Point Approximation

While the above form is hard to evaluate numerically, there is one way of approximating (known as Erlang Fixed Point approximation) the blocking probability by assuming that each link is blocked independently. Then, the route blocking probability

$$L_r \approx 1 - \prod_{j=1}^{J} (1 - B_j)^{A_{jr}}$$

[1]This is a simplified version as it assumes that each call requires exactly one circuit on each link

where B_j is the probability that the link j is full.

As it is assumed that each link is blocked independently, it can be treated as an individual loss system with C_j being the capacity and ν_j being the arrival rate of a call on **link** j. And recall from the previous section,

$$B_j = \text{Erlang}(\nu_j, C_j)$$

and

$$\text{Erlang}(\nu, C) = \frac{\nu^C/C!}{\sum_{n=0}^{C} \nu^n/n!}$$

The idea of EFP approximation is that the link blocking probability B_j can be obtained through iteration. To illustrate how EFP approximation works, I investigated two simple networks and applied EFP approximation to each one of them and tried to compare the blocking probability obtained from EFP with the exact blocking probability. [2]

2.1.2 Ring Network

The ring network I considered is symmetric and has 10 nodes (and therefore 10 links), each link with capacity C being 10. As the network is symmetric, it is easy to see that $B_1 = \ldots = B_j = \ldots = B_{10} = B$. Furthermore, there are only two types of routes allowed in the network. That is, 1-link and 2-adjacent-link routes only. Let the arrival rate of 1-link route calls be ν_1 and that of 2-link route calls be ν_2, then

$$B = \text{Erlang}(\nu, C)$$

where $\nu = \nu_1 + 2\nu_2(1 - B)$.

Figure 1 plots the blocking probabilities, L_1 and L_2, against arrival rate of new calls, and $\nu_1 = \nu_2 =$ arrival rate is assumed. The solid line represents the probability that a call arriving on a 1-link route is blocked (i.e. L_1) and the dashed-dotted line shows the probability that a call arriving on a 2-link route is lost.

As there might still be doubts as to how well EFP approximation works, I checked the relative error of the approximation. I created a Matlab m-file that enumerates statespace of ring networks and calculates the exact blocking probability straight from equation (2). However, due to computation difficulties, I just managed to check a small ring network with $J = 3$ and $C = 3$. Figure (2) plot the relative error of the EFP approximation (around the exact blocking probability) and it can be seen from the graph that for most cases (when the arrival rate is not too small), the relative errors are very close to 0. As this is a **small, linear** networks, which is more likely to have links blocking dependent of each other, it suggests that EFP approximation is a pretty good approximation.

[2] When I say **exact**, I mean without the assumption that the link is blocked independently

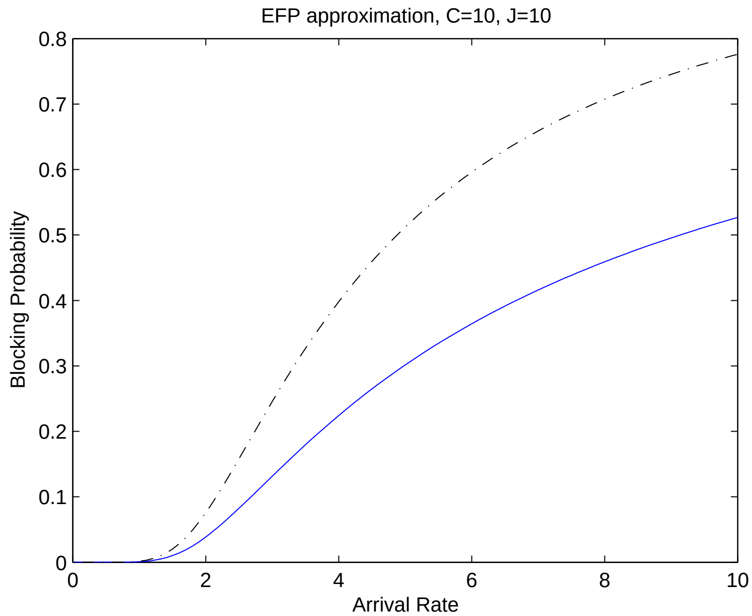

Figure 1: Ring Network with 1-link and 2-link routes

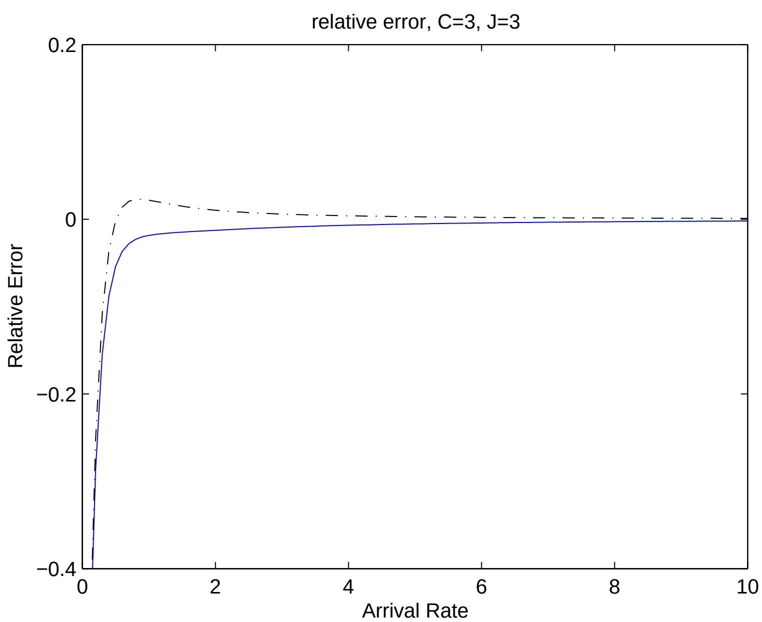

Figure 2: Relative Error of EFP Approx.

In fact, there are algorithms which make the calculation of equation (2) possible. I investigated one outlined in [1]. This algorithm evaluates the normalizing constant of a ring network by evaluating the normalizing constants of a line network.

The line network used here only allows 1-link routes and adjacent 2-link routes. Let $\Psi_K^{(i,j)}$ denote the normalizing constant of a line network with K links (i.e. $K+1$ nodes) and $C_1 = i,\ C_2 = \ldots = C_{K-1} = C,\ C_K = j$, then when the network has no links, the normalizing constant is 1; when the network only has one link then $C_1 = i$, $C_1 = j$ and as one link can only have one capacity, $C_1 = \min(i, j)$.

$$\Psi_0^{(i,j)} = 1,$$

and

$$\Psi_1^{(i,j)} = \sum_{n=0}^{\min(i,j)} \frac{\nu_1^n}{n!}, \qquad \forall\, i, j$$

where ν_1 is the rate that calls arrive on a 1-link routes.

When $K \geq 1$, a $K+2$ node line network can be viewed as adding one node to one end of the network from a $K+1$ node line network. Assume [3] that one node is added to the end of the original network (i.e. after the $K+1$th node), then the new line network with $K+1$ links have capacities $C_1 = i,\ C_2 = \ldots = C_K = C,\ C_{K+1} = j$. Let α be the number of 1-link calls arriving on the $\{K+1\}$ route [4], β be the number of 2-link calls arriving on $\{K, K+1\}$ route, then α, β need to satisfy $\alpha \in [0, j]$ and $\alpha + \beta \in [0, C]$. The normalizing constant of the line network when $K \geq 1$ is therefore,

$$\Psi_{K+1}^{(i,j)} = \sum_{\alpha=0}^{j} \sum_{\beta=0}^{j-\alpha} \frac{\nu_1^\beta}{\beta!} \frac{\nu_2^\alpha}{\alpha!} \Psi_K^{(i,C-\alpha)}$$

where ν_2 is the rate that calls arrive on a 2-link route.

Then, a ring network with K nodes can simply be seen as a $K+1$ node line network but allowing an addition route $\{K, 1\}$, which now makes nodes 1 and $K+1$ the same node. Then the normalizing constant of a K node ring network is

$$\Phi_K = \sum_{n=0}^{C} \frac{\nu_2^n}{n!} \Psi_K^{(C-n,C-n)}$$

Let Φ_K^1 be the normalizing constant of a K node ring network and when there is enough room for a 1-link call to be accepted and Φ_K^2 be the normalizing constant when there is enough circuits for a 2-link call, then

$$\Phi_K^1 = \sum_{n=0}^{C-1} \frac{nu_2^n}{n!} \Psi_K^{(C-n,C-n-1)}$$

[3] This assumption can be made without loss of generality.

[4] Recall that the set of links { } means that a route contains all links that are elements of the set.

and

$$\Phi_K^2 = \sum_{n=0}^{C-1} \frac{nu_2^n}{n!} \Psi_K^{(C-n-1,C-n-1)}$$

I created an m-file in Matlab for this algorithm so that I can evaluate EFP approximation of a larger ring network.

2.1.3 Star Network

The second simple network I examined is called a star network. That is, a network with a central node and J outside nodes, and each pair of outside nodes can create a route via the central node. It is then clear that this network has J links and $\frac{1}{2}J(J-1)$ routes that are all 2-link routes. The network I evaluated is an even simplified version of this which assume symmetric (i.e. the capacities and arrival rate on each link are assumed to be the same).

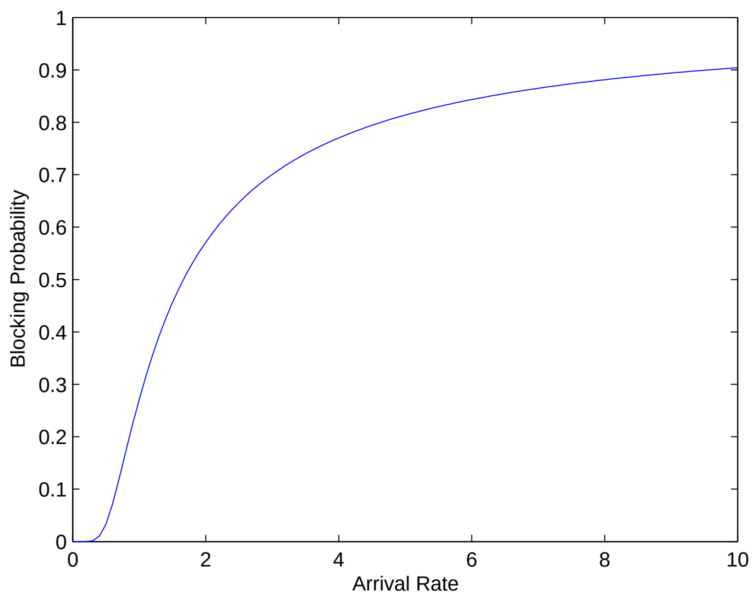

Figure 3: EFP of Star Network

Let ν be the arrival rate of calls on a 2-link route, then the arrival rate of a call on link j is $(J-1)\nu$ and let C be the capacity of a link and B be the blocking probability of a link, then

$$B = \text{Erlang}((J-1)\nu, C)\,,$$

$$L = 1 - (1-B)^2$$

Figure 3 plots the blocking probabilities of any 2-link routes (L) against different 2-link route arrival rates for a star network with $J = 10$ outer nodes.

There is an argument in [1] which says that as a star network is non-linear, then the assumption of the links blocking independently isn't as bad as when the assumption is made in a ring network. Therefore, the EFP approximation should be expected to perform as good as it does in a ring network, if not better. However, I still attempted to check for the relative errors of the approximation by comparing the approximation to the exact blocking probabilities. To do this, I used another algorithm proposed in.

2.2 Alternative Routing

After that, I then turned to look at networks that allow alternative routing. The networks I looked at are complete graphs with J vertices and K edges. That is, a network with J nodes and any two nodes are connected so there are $K = \frac{1}{2}J(J-1)$ links. When a call comes and is blocked on any these K links, it has a second chance of being accepted. That is, it chooses another node at random and gets accepted if both of the two links (connecting this node and the two ending nodes) have free circuits.

In a simplified model, each of the K links contain C circuits and calls arriving on link k as a Poisson process of rate ν. If a call is blocked on the first choice, then any two of the remaining $K-1$ links can be chosen at random and the call is only accepted if both of the two links have free circuits. We also assume that each call ends independently and that the holding times are exponential with mean 1.

The simplified model is very different to the original one as it assumes that

- Holding times are independent, and
- The two chosen alternative links might not be connected at all (i.e. loss of graph structure).

Despite these two "wrong" assumptions, the simplified model is a good approximation to the original network. And it is much simpler to model as it is Markovian. Here is how it can be modelled as a Markov process. Let $n_j^K(t)$ be the number of links that have j circuits in use (out of the total number of K links) at time t, then clearly

$$\sum_{j=0}^{C} n_j^K(t) = K.$$

Now define an operator T_{ij} as

$$T_{ij}n^K = n^K + e_j - e_i \,,$$

where e_i is a vector with 1 on the ith entry and 0 otherwise. Then n^K is a Markov process with transition rates

1. $\mathbf{n}^K \to T_{j,j+1}\mathbf{n}^K$, one arrival on link with j circuits in use at rate

$$\nu n_j^K, \qquad j = 0, 1, \ldots, C-1$$

2. $\mathbf{n}^K \to T_{j,j-1}\mathbf{n}^K$, one existing call ends at rate

$$j n_j^K, \qquad j = 1, 2, \ldots, C$$

3. $\mathbf{n}^K \to T_{i,i+1}T_{j,j+1}\mathbf{n}^K$, at rate

$$\nu n_C^K \frac{n_i^K n_j^K}{\binom{K-1}{2}}, \qquad i > j, i, j = 0, 1, \ldots, C-1$$

4. $\mathbf{n}^K \to T_{j,j+1}^2\mathbf{n}^K$, at rate

$$\nu n_C^K \frac{\binom{n_j^K}{2}}{\binom{K-1}{2}}, \qquad j = 0, 1, \ldots, C-1$$

Define $x_j^K = n_j^K/K$, then $\mathbf{x}^K$ lies in the simplex

$$\Delta = \{\mathbf{x}^K \in R_+^{C+1} : \sum_{i=0}^{C} x_i^K = 1\}$$

.

Theorem 1 *When $\mathbf{x}(\cdot)$ is the unique solution to the following set of equations, then if $\mathbf{x}^K(0) \Rightarrow \mathbf{x}(0)$ then $\mathbf{x}^K(\cdot) \Rightarrow \mathbf{x}(\cdot)$*

$$\frac{d}{dt}x_0(t) = x_1(t) - (\nu + \lambda(t))x_0(t) \tag{3}$$

$$\frac{d}{dt}x_j(t) = (\nu + \lambda(t))x_{j-1}(t) - (\nu + \lambda(t) + j)x_j(t) + (j+1)x_{j+1}(t), \qquad 1 \le j \le C-1 \tag{4}$$

$$\frac{d}{dt}x_C(t) = (\nu + \lambda(t))x_{C-1}(t) - Cx_C(t) \tag{5}$$

where

$$\lambda(t) = 2\nu x_C(t)(1 - x_C(t))$$

Thus, $\mathbf{x} = (x_0, x_1, \ldots, x_C) \in \Delta$ is a fixed point of the equations if and only if

$$(n+1)x_{n+1} = (\nu + \lambda)x_n, \qquad n = 0, 1, \ldots, C-1$$

Therefore, a fixed point $\mathbf{x}$ is clearly of the form

$$x_j = \frac{\xi^j}{j!}\left(\sum_{i=0}^{C}\frac{\xi^i}{i!}\right)^{-1} \qquad j = 0, 1, \ldots, C$$

where

$$\xi = \nu + \lambda = \nu + 2\nu x_C(1 - x_C)$$

and

$$x_C = E(\xi, C) = \frac{\xi^C}{C!}\left(\sum_{i=0}^{C}\frac{\xi^i}{i!}\right)^{-1}$$

x_C is the proportion of links that are full by definition so it approximates the link blocking probability (B). Then,

$$B = E(\xi, C) = E(\nu + 2\nu B(1 - B), C)$$

There are also networks that allow for more than one retries. In such a case, the rate of arrival on each link, ξ is different from above and is

$$\xi = \nu + 2\nu B\{1 - [1 - (1 - B)^2]^r\}$$

where r is the number of retries (notice how this returns to the above result when $r = 1$), and

$$B = E(\xi, C)$$

However, for large C values, it is difficult to calculate a numerical value for quation 6 as both the denominator and the numerator are so large that Matlab treat them as $+\infty$ and therefore the ratio of the two is not a number. However, by rearranging this to equation 7, it becomes less a problem. First of all, as the bistability problem we are examining only happens when the load (ξ/C) is near 1, then the numerator is still greater than 0 for C very large. And when C is really large, terms with small i values in the denominator tend to 0 but this is okay because some terms are still non-zero and therefore the sum of al these terms (i.e. the denominator) is also non-zero.

$$\text{Erlang}(\xi, C) = \frac{\frac{\xi^C}{C!}}{\sum_{i=0}^{C}\frac{\xi^i}{i!}} \tag{6}$$

$$= \frac{(\frac{\xi}{C})^C}{\sum_{i=0}^{C}(\frac{\xi}{C})^i\frac{C(C-1)\cdots(i+1)}{C^{C-i}}} \tag{7}$$

Figure 4 plots the blocking probabilities of one link when there is only 1 retry (i.e. after the attempt of the direct route, it gets one chance to choose any two alternative links to form a route, if any of the links on this choice is full, then the call is blocked) and when the capacity for each link is 120 or 1000 respectively. It can be seen that when the capacity of the link increases, the curvature of the curve increases, which leads to a greater region of bistability. Moreover, with any given ν/C, the link blocking probability of the network with capacity 1000 is always lower than that of the network with capacity 120.

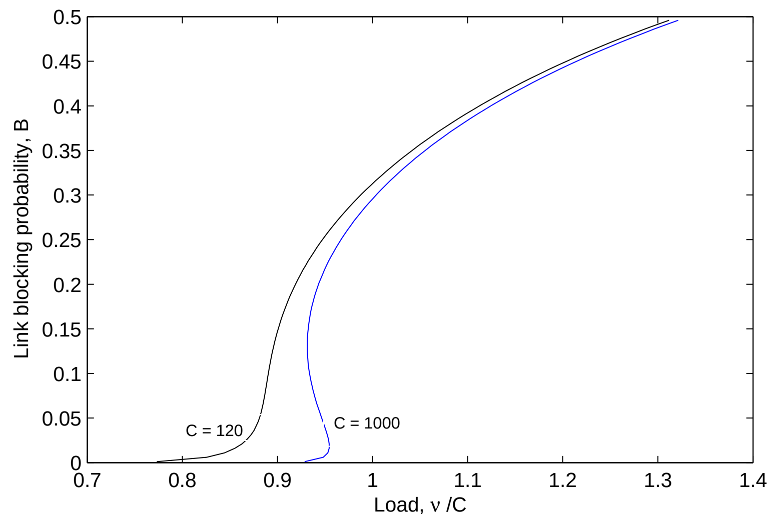

Figure 4: Blocking probability with 1 retries

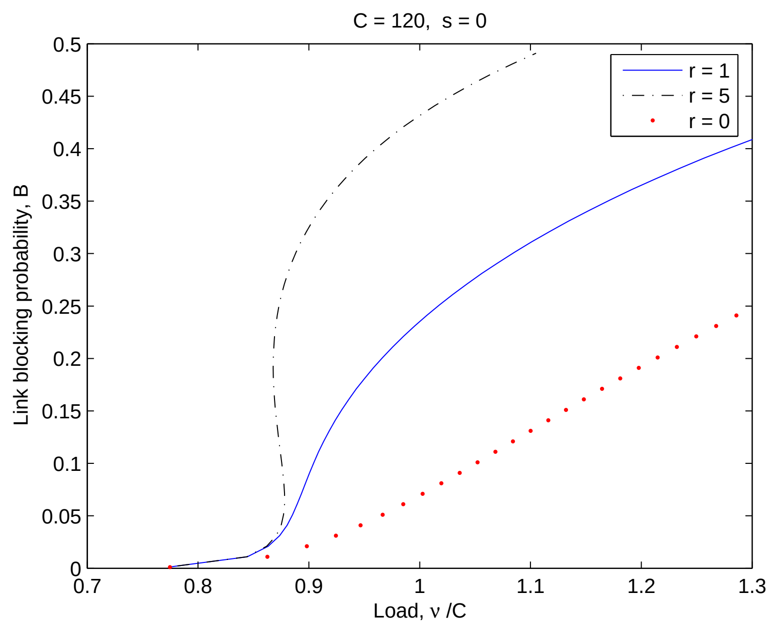

Figure 5: When capacity is 120 and retries are 0, 1 and 5

When the concept of alternative routes is introduced, it is in hopes that the performance of the network will be better. As this perfomance is measured in terms of its blocking probabilities, we should now look at how the number of retries affect the link blocking probabilities and the call blocking probabilities.

Figure 5 plots the link blocking probabilities when the number of retries is 0, 1 or 5 respectively. From the figure, it suggests that with the increase of the number of retries, the region for bistability increases and not only that, the link blocking probability increases as well. Intuitively, an increase the number of retries means a greater demand for (or a more frequent arrival of calls on) any link. Given unchanged capacity, one would expect that the link blocking probabilities to increase. This increase in link blocking probabiity is bad as it means that each link is more often running at full capacity. However, it is not necessarily that bad to a customer as he/she is more interested in the call blocking probabilities. That is, when a customer arrives to make a call, he/she is more interested in whether the call finally goes through rather than knowing whether it goes through a direct route or whether it takes several retries.

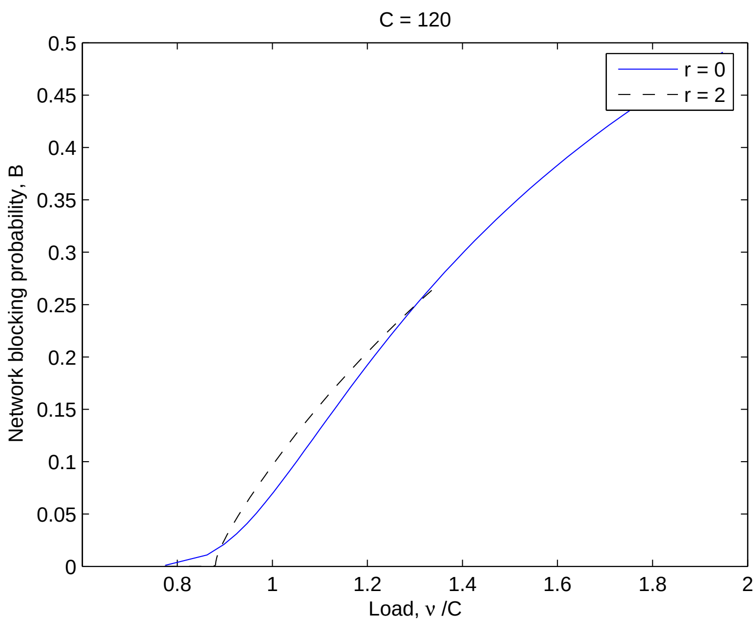

Figure 6: Call/Network blocking probability

Due to the reasons discussed above, I then looked at call blocking probabilities (or network blocking probability). Figure 6 shows that when the number of retries is allowed to rise from 0 to 2, for some values of ν/C, the network blocking probabilities increase. If this is always the case, then we shouldn't allow any retries to exist because it not only increases the link blocking probability but also increases the network blocking probability. But take a look at figure 7, when the number of retries is 5, then the network blocking probability curve is always below that of no retries. This suggests that although an increase in the number of retries allowed increases the link blocking probability all the time, when it reaches a particular value, it decreases the network blocking probability and this is, as I discussed above, the primary interest of any customer.

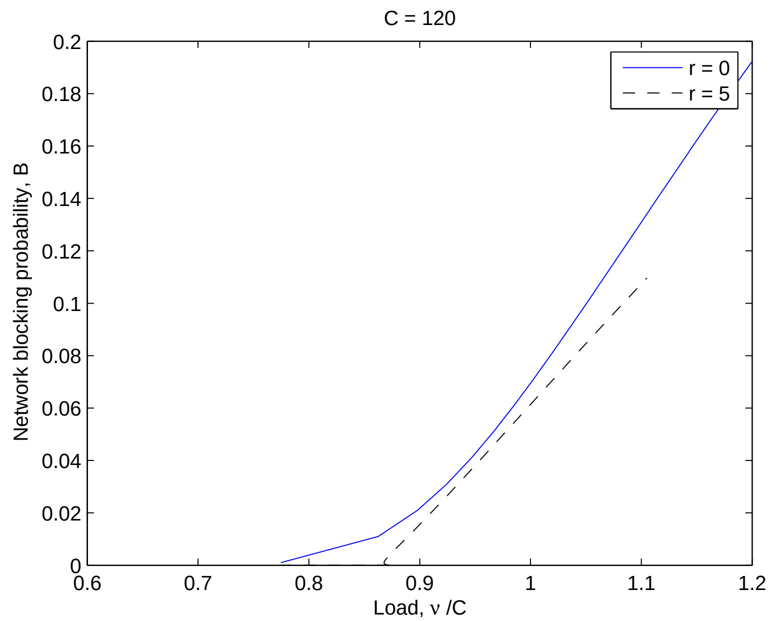

Figure 7: Call/Network blocking probability

As this above result is different from the one stated in [2] (labelled here as GHK paper), further investigation should be taken before any more inferences may be made.

2.2.1 Trunk Reservation

Finally, I briefly looked at how trunk reservation might improve the preformance of a network. Let s be the number of circuits reserved in a link, then clearly $s \in [0, C]$. Following the results stated in GHK paper that says:

A fixed point $\mathbf{x} = (x_0, x_1, \ldots, x_C) \in \Delta$ of this network satisfies

$$(j+1)x_{j+1} = (\nu + \lambda)x_j \qquad j = 0, 1, \ldots, C - s - 1 \tag{8}$$

$$(j+1)x_{j+1} = \nu x_j \qquad j = C - s, \ldots, C - 1 \tag{9}$$

where

$$\lambda = 2\nu B_1 (1 - B_2)^{-1}\{1 - [1 - (1 - B_2)^2]^r\}$$

$$B_1 = x_C, \quad B_2 = \sum_{i=C-s}^{C} x_i$$

Figure 8 shows that when the circuits reserved increases, there is a decrease in link blocking probability and a decrease in the region of bistability. This is what I expected as if s goes

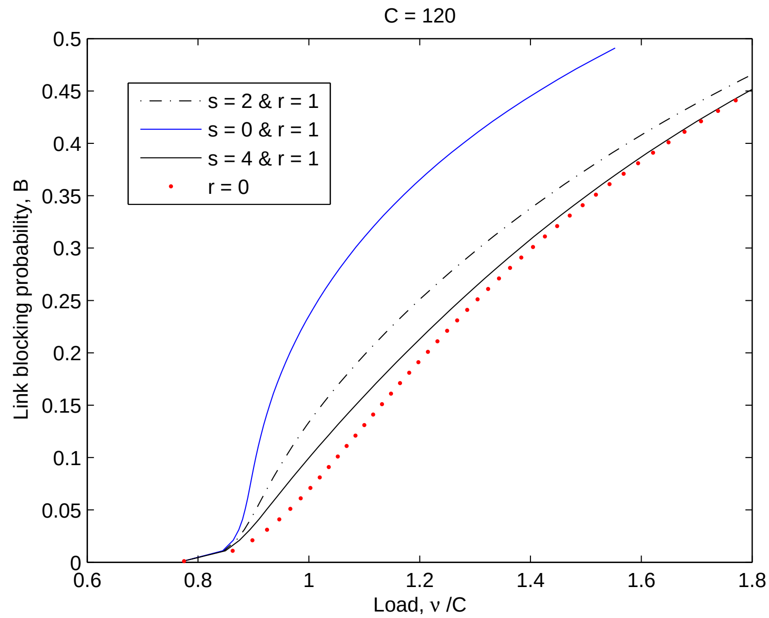

Figure 8: Trunk reservation

up to C, then it returns back to the case when there is only direct routes available. So on the graph, when s increases, the curve gets closer and closer to the curve when there is no alternative routes.

3 Conclusion

This is just a brief summary of what I have been doing for the past six weeks for my summer vacation scholarship. Due to time constraints, there are still lots of areas that I wish to explore but haven't got time to. Here is a list of some.

1. Although the algorithm for the ring network to calculate the exact blocking probability works, it might still be a good idea to check the m-file as it is not working as accurate as I expect.

2. There are several papers on star network and its exact blocking probability. It might be worth exploring further.

3. I would like to reproduce some of the trajectory plots in GHK paper.

4. It would be good to be able to use the 1-dimensional diffusion approximation mentioned in GHK paper.

5. As mentioned above, the result of the network blocking probability I got is different from the one stated in GHK paper, I need to double check my work to ensure that the plots I used to draw the conclusion are actually the right ones.

6. The last bit and hardest bit will be to think about the manifold problem.

References

[1] Bebbington, M., Pollett, P.K. and Ziedins, I. (1998) Two-link approximation schemes for linear loss networks without controls. *Journal of Korean Mathematical Society* 35, 539-557.

[2] Gibbens, R.J., Hunt, P.J. and Kelly, F.P. (1989) Bistability in communication networks. In *Disorder in physical systems,* ed. G.R. Grimmett and D.J.A. Walsh. Oxford University Press, Oxford. pp. 113-128