Queuing (Trunking) Theory

Introduction

Queuing theory is the theory of waiting in line. It answers questions such as: If N customers are trying to randomly access a system that can serve M at a time, what is the probability that someone will be denied service? The theory has wide applications, from determining the number of check-out counters at a supermarket, to providing guidelines for the required capacity of a power system, to telling an ISP how many dial-up modems to have. It's important for us because it tells us how many customers we can serve with a given number of phone channels.

Theory

Erlang studied the problem of telephone system congestion and published his analysis in 1917. He assumed that the probability of a call being initiated during a short time interval dt is λdt where λ is some constant. In other words he assumed that calls are initiated randomly at a uniform (on average) rate. The parameter λ is the average number of calls per unit time made by all users. If there are U users in the system and each user makes λ_1 calls per unit time then $\lambda = U\lambda_1$.

Erlang also assumed that the probability that any particular call would terminate during a short time interval dt is μdt where μ is some other constant. You can show that this means that the probability for a phone call have a duration between t and $t+dt$ is given by $\mu e^{-\mu t}dt$. You can also show that the probability that the elapsed time between phone calls is be between t and $t+dt$ is given by $\lambda e^{-\lambda t}dt$. (See Appendix.) These assumptions are generally verified by observed telephone usage statistics.

With these assumptions, the average duration of a phone call is

$$\langle t \rangle = \int_0^\infty t\mu e^{-\mu t}dt = \frac{1}{\mu} \equiv H \tag{15.1}$$

H is called the *holding time*.

Say we have a system with C phone channels. Assume that $p_k(t)$ is the probability that k of these channels are occupied at time t, for $0 \le k \le C$. Let's consider the question: What is the probability that there are n calls in the system at time $t+dt$? We will denote this by $p_n(t+dt)$. Since dt is a very short time interval we will consider that at most a single event can occur during dt. Then there are three ways to arrive at having n calls. The first is that there were already n calls and no calls were initiated or terminated. The probability of a call being initiated is λdt. The probability of a particular call being terminated is μdt, so the probability of any one of n calls being terminated is $n\mu dt$ (provided μdt is very small). Therefore, the probability of neither happening is $1-(\lambda+n\mu)dt$. This scenario therefore has a probability of $p_n(t)[1-(\lambda+n\mu)dt]$.

The second possibility is that there were $n-1$ calls at time t and a new one was initiated. This has a probability of $p_{n-1}(t)\lambda dt$. The third is that there were $n+1$ calls at time t and one was terminated. This has a probability of $p_{n+1}(t)(n+1)\mu dt$. Summing these up we arrive at

$$p_n(t+dt) = p_n(t)[1-(\lambda+n\mu)dt] + p_{n-1}(t)\lambda dt + p_{n+1}(t)(n+1)\mu dt \tag{15.2}$$

Now consider a steady-state solution where the probabilities don't change with time, i.e., $p_n(t) = p_n$. This requires, for $0 < n < C$,

$$p_n = p_n[1-(\lambda+n\mu)dt] + p_{n-1}\lambda dt + p_{n+1}(n+1)\mu dt \tag{15.3}$$

or

$$(n+1)p_{n+1} = \rho p_n + np_n - \rho p_{n-1} \tag{15.4}$$

where $\rho = \lambda/\mu$. For the cases $n = 0$ and $n = C$ the equations are, respectively

$$\begin{aligned} p_1 &= \rho p_0 \\ 0 &= Cp_C - \rho p_{C-1} \end{aligned} \tag{15.5}$$

because in the first case we cannot have -1 calls, and in the second we cannot have $C+1$ calls. A solution to (15.4) and (15.5) is

$$p_n = \frac{\rho^n}{n!} \tag{15.6}$$

(plug it in and verify this). However, since there is a 100% probability that some number of channels between 0 and C will be occupied we must have

$$\sum_{n=0}^{C} p_n = 1 \tag{15.7}$$

This requires us to normalize our solution as follows:

$$p_n = \frac{\dfrac{\rho^n}{n!}}{\displaystyle\sum_{k=0}^{C}\frac{\rho^k}{k!}} \tag{15.8}$$

Now, the probability that a user will experience a blocked call is the same as the probability that all C channels are occupied, or

$$PB = \frac{\frac{\rho^C}{C!}}{\sum_{k=0}^{C}\frac{\rho^k}{k!}} \tag{15.9}$$

where *PB* is the probability of blocking, and ρ is called the *offered traffic intensity* and has units of *Erlangs*.

Evaluating this equation is problematic for large values of *C* because you have the problem of dividing huge numbers. A more useful form for calculation is

$$\frac{1}{PB} = 1 + \frac{C}{\rho} + \frac{C}{\rho}\frac{C-1}{\rho} + \frac{C}{\rho}\frac{C-1}{\rho}\frac{C-2}{\rho} + \cdots \tag{15.10}$$

Each term can be obtained from the previous via multiplication by the ratio of two reasonably sized numbers and so we avoid the problem mentioned above. The following Matlab subroutine implements this.

```
function p = PB(C,rho)
sum = 1;
x = 1;
for i=C:-1:1
   x = x*(i/rho);
   sum = sum+x;
end
p = 1/sum;
```

For example, with 40 channels and a traffic intensity of 35 Erlangs, we find a *PB* of 5.4%. Typically we want to invert this equation to find ρ when given values of *C* and *PB*. The following Matlab code is a simple implementation of this.

```
function rho = erlangs(C,pb)
rho = C; %rho will generally be less than C, so we'll work downward.
%but just in case we need rho>C, we check for that first.
while (PB(C,rho)<=pb)
    rho = rho+1;
end
%now find integer part that gives PB just higher than desired
while PB(C,rho)>pb
   rho = rho-1;
end
rho = rho+1;
%find first decimal place that gives PB just higher than desired
while PB(C,rho)>pb
   rho = rho-0.1;
end
rho = rho+0.1;
%find first second place that gives PB just lower than desired
while PB(C,rho)>pb
   rho = rho-0.01;
end
```

As an example, say we have 40 channels and we want a *PB* of 2%. Then we find that we can support an offered traffic intensity of 31 Erlangs. Since 2% of the offered traffic is blocked, we end up with a *carried traffic intensity* that is 98% of this value.

What I'm calling "probability of blocking" is often called "grade of service" or GOS. So a system might be said to offer a "2% grade of service." Personally I find this terminology a bit confusing. From the sound of it you might expect that you'd want the highest possible "grade of service." After all, who wants a low grade of anything – imagine if your phone company started charging you more because they "are now offering you a lower grade of service?" But a higher GOS means that it will be harder to place calls. Maybe "irritation of service" would be more appropriate. In this course we will use "probability of blocking" as it is more to the point.

Simulation

You can easily simulate a trunked-channel system. Step through time at intervals of *dt*. At each step look at a random number *x* uniformly distributed over [0,1]. If $x < \lambda dt$ then assume a new call is attempted. If all channels are full then the new call is blocked. Otherwise we occupy an empty channel. Then for each call in progress look at a new random number *x*. If $x < \mu dt$ "hang up" the call and clear the channel. Continuing on in this manner we can keep track of the number of channels occupied as a function of time and the fraction of attempted calls that are blocked. The following Matlab code implements this.

```
%trunksim.m performs a trunked channel simulation

C = 10; %number of trunked channels
H = 3; %average call duration in minutes
lambda = 2; %average number of calls per minute
tfin = 100; %number of minutes to simulate
dt = 1; %time step in seconds

tfin = tfin*60; %convert to seconds
Nattempt = 0;
Nblocked = 0;
lambda = lambda/60; %convert to calls per second
H = H*60; %convert to seconds
mu = 1/H;
M = 0; %M will be the # of channels currently in use
t = 0;
k = 1;

while (t<tfin)
   % begin by seeing if some wants to make a call
   x = rand(1); %"flip a continuous coin"
   if (x<lambda*dt) %condition for call attempted
      Nattempt = Nattempt+1; %keep track of # of calls
      if (M==C) %call blocked, all channels are full
         Nblocked = Nblocked+1; %keep track of number of blocks
      else %otherwise we use up a new channel
         M = M+1;
      end
   end
```

```
    % now see how many of the current users want to hang up
    hangups = 0;
    x = rand(1,M); %flip M coins
    for i=1:M
          if (x(i)<mu*dt) %condition for call terminated
          hangups = hangups+1;
          end
    end
    M = M-hangups; %free up channels where user hung up
    t = t+dt;
    occupied(k) = M; %keep track of traffic vs. time
    k = k+1;
end
plot(occupied);
Nattempt
Nblocked
GOS = 100*(Nblocked/Nattempt)
axis([0 6000 0 10]);
```

Simulation results are shown in Fig. 15.1. We see that the number of channels occupied fluctuates considerably.

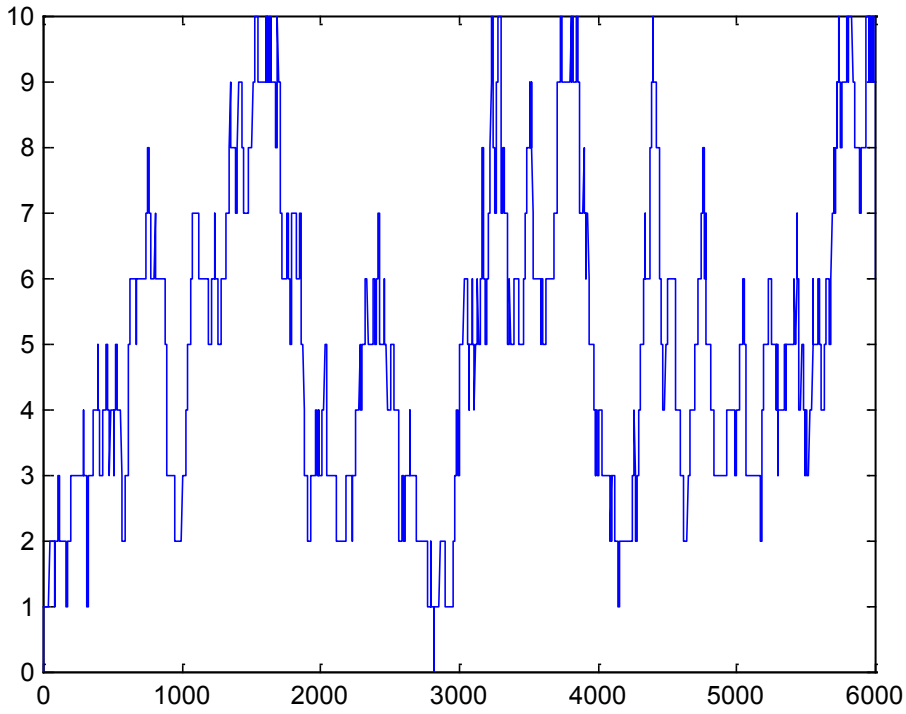

Figure 15.1: Simulation results. Number of occupied channels (vertical axis) vs. seconds of time (horizontal axis). A blocked call results when all 10 channels are occupied and someone tries to make a new call. In this simulation $PB \approx 0.02$.

Using the PB Equation in Cellular Design

Note that

$$\rho = \frac{\lambda}{\mu} = \lambda_1 HU \tag{15.11}$$

The dimensionless product $\lambda_1 H$ is the fraction of the time that an individual user wants to be on the phone. For example, if the average person wants to make 2 calls per hour and each lasts on average 5 minutes, then $\lambda_1 H = 1/6$ meaning that the typical user wants to spend 1/6 of her time on the phone. If $U = 600$ users then $\rho = 100$ Erlangs of *offered* traffic. If $PB = 0.02$ then the system will support 98 Erlangs of *carried* traffic.

This means that 600 people who want to use their phones 1/6 of the time produce the same carried traffic as 10 people who use their phones all of the time. *Thus the Erlangs of carried traffic intensity can be considered as the equivalent number of full-time phone users or full-time phone calls.*

Erlangs (of carried traffic) are what a service provider makes money on, not the number of physical channels they have. For example, if a cell site supports 50 Erlangs of *carried* traffic, and the billing rate for calls is \$0.10/min, then the cell site generates revenue at the rate \$5/minute, that is

Revenue rate = (Erlangs of carried traffic) x (billing rate)

If you want to make more money you have to either increase your billing rate (your price per Erlang) or increase your traffic intensity (your number of Erlangs). Increasing your billing rate will probably drive your customers to another company. So, increasing your traffic intensity is more promising. How do you increase your traffic intensity ρ? You need to increase at least one of the three factors in (15.11). One approach is the increase the number of users *U*. Another is to encourage users to use their phones more often, i.e., to increase λ_1, and/or to talk longer, i.e, increase *H*. However as you increase ρ, for a fixed number of channels *C*, the *PB* increases. As it does more and more users will get busy signals and become irritated with your service. If a competitor offers a lower *PB* they are likely to switch service providers. Consequently to obtain an increase in traffic intensity typically requires some engineering improvements to avoid an increase in *PB*.

Using the *PB* concept in cellular design usually goes as follows:

1. Determine the number of physical channels available per cell.
2. Find the offered traffic intensity per cell that gives an acceptable *PB*.

3. For estimated phone utilization $\lambda_1 H$, translate this into the number of customers you can serve per cell.

4. For a given user density this determines the desired cell size.

Here's an example.

Example 15.1

1. We have 200 total channels available in an $N = 4$ reuse pattern. Therefore we have 50 channels per cell available.

2. We want to offer service with $PB = 2\%$. Calculating the traffic intensity possible for 50 channels with $PB = 2\%$ we find that we can support 40 Erlangs of offered traffic.

3. Let's assume phone utilization is $\lambda_1 H = 0.01$, i.e., customers spend 1% of their time on the phone. Then $40 = 0.01U$ gives $U = 4000$ users per cell.

4. Assume that there is one customer per $1000\,\text{m}^2$. This is a density of $1000\,\text{km}^{-2}$ Then our cell can serve $4000/1000 = 4\,\text{km}^2$. Since the area of a cell is $2.6R^2$ this gives $R = 1.24\,\text{km}$ for our cell radius.

Trunking Efficiency

Trunking efficiency is a measure of channel utilization at a given *PB*. An efficiency of 100% would mean that every channel is always in use, an efficiency of 50% that channels are used ½ the time, on average, and so on. The efficiency is simply the number of carried Erlangs per channel, or $(1 - PB)\rho / C$. In Fig. 15.1 this is plotted for $PB = 2\%$.

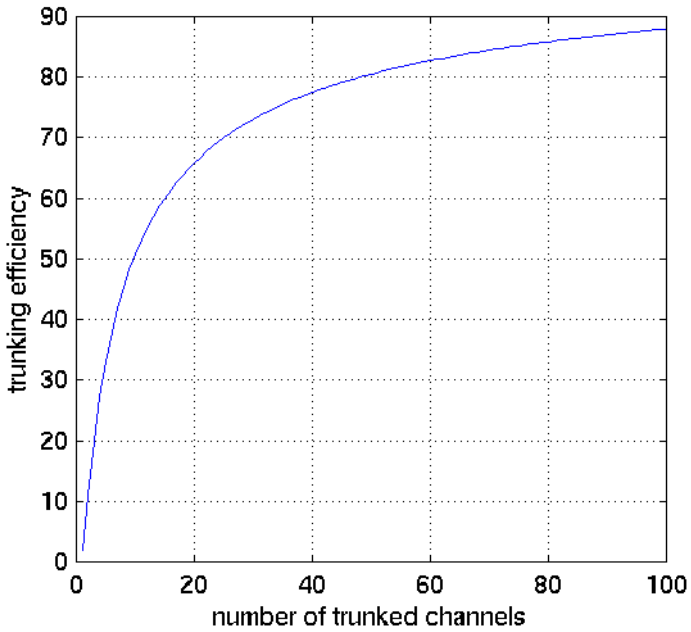

Figure 14.1.Trunking efficiency vs. number of channels for PB=2%.

Trunking efficiency increases with the number of channels. It drops off dramatically when C gets small. We need to have a few channels unused to allow new calls to be made. When C is large these few "buffer" channels are a small fraction of the total, but when C is small they are a large fraction. Hence efficiency increases with C.

Consider the simulation results in Fig. 15.1 where $C = 10$. The trunking efficiency is the fraction of the graph's area that falls under the usage curve. Fig. 15.2 shows that for 10 channels the efficiency should be about 50%. We can see that indeed about ½ the area of the Fig. 15.1 graph falls under the usage curve.

References

1. Saaty, Thomas L., *Elements of Queuing Theory With Applications*. Dover 1983. ISBN 0-486-64553-3. See in particular section 14-2 "Applications to Telephone Congestion."

Appendix

Assume your have made a phone call. You have an N-sided die. On one side of the die is written "hang up" while on the $N-1$ other sides is written, "stay on the line." The probability that "hang up" will come up is $1/N$. Otherwise you stay on the line. After Δt seconds you roll the die again. If "hang up" comes up then you hang up. If not then you stay on for another Δt seconds, and so on. What is the probability that you will stay on the line $n\Delta t$ seconds and hang up by $(n+1)\Delta t$ seconds?

Define μ so that $\mu\Delta t = 1/N$. Then the probability that you will hang up at any particular roll of the die is $\mu\Delta t$ while the probability that you will stay on the line is $1-\mu\Delta t$. For your call to last $n\Delta t$ seconds you need to roll "stay on the line" for n rolls and then roll "hang up." The probability of this occurring is $(1-\mu\Delta t)^n \mu\Delta t$ and this is the probability that your phone call will last $t = n\Delta t$ seconds but terminate by $t = (n+1)\Delta t$ seconds. If $\mu\Delta t$ is very small then $e^{-\mu\Delta t} \to 1-\mu\Delta t$ and we can write $(1-\mu\Delta t)^n \mu\Delta t = e^{-\mu n\Delta t}\mu\Delta t = \mu e^{-\mu t}\Delta t$. With $\Delta t \to dt$ we arrive at the probability that your phone call will last t seconds but terminate by $t+dt$ seconds: $\mu e^{-\mu t} dt$.

For a given number of channels (C) this table gives the offered traffic intensity (rho) that results in a PB of 2%.

C	rho	C	rho	C	rho	C	rho	C	rho
5	1.65	6	2.27	7	2.93	8	3.62	9	4.34
10	5.08	11	5.84	12	6.61	13	7.40	14	8.20
15	9.00	16	9.82	17	10.65	18	11.49	19	12.33
20	13.18	21	14.03	22	14.89	23	15.76	24	16.63
25	17.50	26	18.38	27	19.26	28	20.15	29	21.03
30	21.93	31	22.82	32	23.72	33	24.62	34	25.52
35	26.43	36	27.34	37	28.25	38	29.16	39	30.08
40	30.99	41	31.91	42	32.83	43	33.75	44	34.68
45	35.60	46	36.53	47	37.46	48	38.39	49	39.32
50	40.25	51	41.18	52	42.12	53	43.05	54	43.99
55	44.93	56	45.87	57	46.81	58	47.75	59	48.70
60	49.64	61	50.58	62	51.53	63	52.48	64	53.42
65	54.37	66	55.32	67	56.27	68	57.22	69	58.17
70	59.12	71	60.08	72	61.03	73	61.98	74	62.94
75	63.90	76	64.85	77	65.81	78	66.77	79	67.72
80	68.68	81	69.64	82	70.60	83	71.56	84	72.52
85	73.49	86	74.45	87	75.41	88	76.37	89	77.34
90	78.30	91	79.27	92	80.23	93	81.20	94	82.16
95	83.13	96	84.10	97	85.06	98	86.03	99	87.00
100	87.97	101	88.94	102	89.91	103	90.87	104	91.85
105	92.82	106	93.79	107	94.76	108	95.73	109	96.70

For a given number of channels (C) this table gives the offered traffic intensity (rho) that results in a PB of 5%.

C	rho	C	rho	C	rho	C	rho	C	rho
5	2.21	6	2.96	7	3.73	8	4.54	9	5.37
10	6.21	11	7.07	12	7.95	13	8.83	14	9.72
15	10.63	16	11.54	17	12.46	18	13.38	19	14.31
20	15.24	21	16.18	22	17.13	23	18.07	24	19.03
25	19.98	26	20.94	27	21.90	28	22.86	29	23.83
30	24.80	31	25.77	32	26.74	33	27.72	34	28.69
35	29.67	36	30.65	37	31.63	38	32.62	39	33.60
40	34.59	41	35.58	42	36.57	43	37.56	44	38.55
45	39.55	46	40.54	47	41.54	48	42.53	49	43.53
50	44.53	51	45.53	52	46.53	53	47.53	54	48.53
55	49.53	56	50.54	57	51.54	58	52.55	59	53.55
60	54.56	61	55.57	62	56.58	63	57.58	64	58.59
65	59.60	66	60.61	67	61.63	68	62.64	69	63.65
70	64.66	71	65.67	72	66.69	73	67.70	74	68.72
75	69.73	76	70.75	77	71.76	78	72.78	79	73.80
80	74.81	81	75.83	82	76.85	83	77.87	84	78.89
85	79.91	86	80.93	87	81.95	88	82.97	89	83.99
90	85.01	91	86.03	92	87.05	93	88.07	94	89.10
95	90.12	96	91.14	97	92.16	98	93.19	99	94.21
100	95.24	101	96.26	102	97.28	103	98.31	104	99.33
105	100.36	106	101.38	107	102.41	108	103.44	109	104.46