# Comp GZ02 (Network Performance)

# Model answers 2006/2007

---

$\boxed{1}$ (a). The arrival process is a Poisson process of rate $\lambda$.

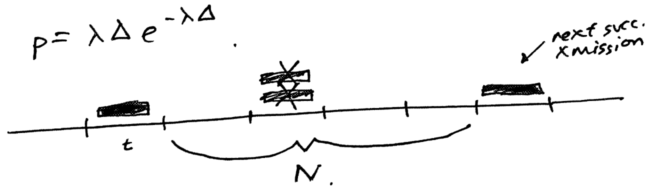
2/33 Let $A_t$ be the number of arrivals in an interval of length $t$; then $A_t$ has a Poisson distribution with mean $\lambda t$.

We want:

$$\mathbb{P}(A_\Delta = 1) = \frac{(\lambda\Delta)^1 e^{-\lambda\Delta}}{1!} = \lambda\Delta e^{-\lambda\Delta}.$$

(b). Let $p = \lambda\Delta e^{-\lambda\Delta}$.

7/33




$$N = \begin{cases} 0 & \text{if in timeslot } t+1 \text{ there is a succ. xmit} & \text{which has prob. } p \\ 1 & \text{if in timeslot } t+1 \text{ there is no succ. xmit}, \; t+2 \text{ there is a succ. xmit} & \text{which has prob. } (1-p)p \\ \vdots & & \vdots \\ r & \text{if in timeslots } t+1, \dots, t+r \text{ there is no succ. xmit}, \; t+r+1 \text{ there is a succ. xmit} & \text{which has prob. } (1-p)^r p. \end{cases}$$

Note that $N+1$ is a standard geometric random variable with parameter $p$; thus the mean value of $N+1$ is $\frac{1}{p}$; thus the mean value of $N$ is $n = \frac{1}{p} - 1$.

[see Ex. Sheet 2, Q9.

(c) 5/33

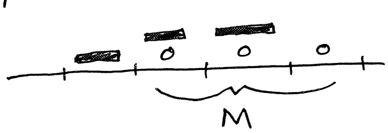
$$\theta = \frac{1}{\Delta(1+n)} = \frac{1}{\Delta(\frac{1}{p} - 1 + 1)} = \frac{p}{\Delta} = \lambda e^{-\lambda\Delta}$$

[Recall this expression from the first model for shared access networks.

(d). 2/33

$$\mathbb{P}(A_\Delta \geq 1) = 1 - \mathbb{P}(A_\Delta = 0) = 1 - \frac{\lambda\Delta^0 e^{-\lambda\Delta}}{0!} = 1 - e^{-\lambda\Delta}$$

(e). 6/33 Let $q = e^{-\lambda\Delta}$ be the probability that the repeat is successful, ie that no new packets arrive.




$\mathbb{P}(M = r) = (1-q)^{r-1} q$, ie $M$ is a standard geometric random variable, so it has mean value $m = \frac{1}{q}$.

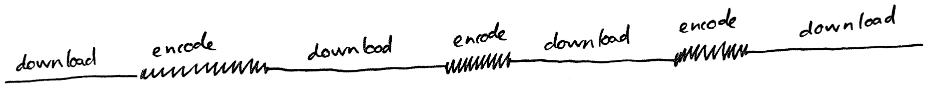
(f) 5/33 Exactly the same $(n)$ as in part (b).

(g). 6/33

$$\theta_{rep} = \frac{1}{\Delta(1+m+n)} = \frac{\lambda e^{-\lambda\Delta}}{\lambda\Delta + 1} \quad \text{and} \quad \text{efficiency} = \frac{\theta_{rep}}{\theta} = \frac{1}{\lambda\Delta + 1}.$$

2. This is a question about Markov chains (see the keywords "transition rates" and "equilibrium distribution" in the question).
It is furthermore a question about job models (see the keyword "processor sharing").

(a). He is running $M$ copies of the program concurrently.
3/33 Each copy repeats the cycle indefinitely:

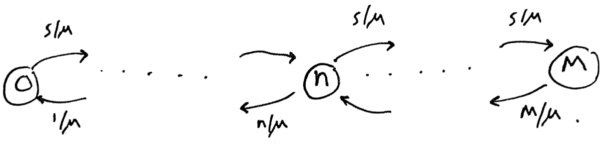



At any point in time $t$, $N_t$ copies are downloading and $M - N_t$ are encoding.
Clearly $0 \le N_t \le M$.
Whenever a download finishes, $N_t$ decreases by 1;
whenever an encode finishes, $N_t$ increases by 1 (since that thread starts downloading).


(b). Suppose there are $n$ downloads at some instant in time.
18/33 Each download has duration $\sim \text{Exp}(\frac{1}{\mu})$, and there are $n$ of them;
by the minimums property the time until the first of these $n$ finishes is $\sim \text{Exp}(\frac{n}{\mu})$.
[ see the Erlang model.

Suppose there are $n$ downloads and $M-n$ encodes at some instant in time.
Each encode would take $\sim \text{Exp}(\frac{s}{\mu})$ if it had exclusive access to the CPU.
But the CPU is shared between the $M-n$ encodes, so each takes $\sim \text{Exp}\left(\frac{s/(M-n)}{\mu}\right)$.
By the minimums property, the time until the first of these $M-n$ finishes is $\sim \text{Exp}(\frac{s}{\mu})$.
[ See the TCP processor sharing model.

These calculations give us the transition rates




(c). This is exactly the transition rate diagram for the Erlang model
8/33 with traffic intensity $\rho = \frac{s/\mu}{1/\mu} = s$. So $\mathbb{P}(N_t = n) = \frac{\rho^n/n!}{\sum_{i=0}^{M} \rho^i/i!}$

(d). The CPU is underutilized when there are no threads encoding (ie $N_t = M$),
4/33 according to the question. By the Ergodic result for Markov chains,
the fraction of time spent in this state is just

$$\mathbb{P}(N_t = M) = \frac{\rho^M/M!}{\sum_{i=0}^{M} \rho^i/i!}.$$

3. This is a question about the fixed point method for TCP (see the keyword "iterative procedure").

[See the network setup in the "Web cache placement" example.
See also the page in the notes describing fixed points for TCP.

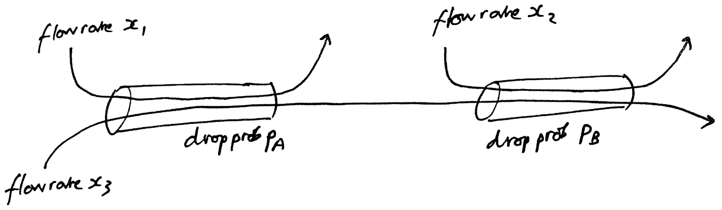
(a). The TCP throughput equation is $x = \dfrac{\sqrt{2}}{RTT\sqrt{p}}$
4/33 where $x$ is the throughput of a single TCP flow (in pkts/sec), RTT is its round trip time (in sec), and $p$ is the packet drop probability.

(b). The question gives capacities in Mb/s; to fit in with TCP throughput equation,
2/33 we should convert to pkt/sec. A reasonable guess is that $1\text{pkt} \approx 1\text{kByte}$.
Thus the link capacities are

$$C_A = 1\ \text{Mb/s} \approx 120\ \text{pkt/sec.}$$

$$C_B = 3\ \text{Mb/s} \approx 360\ \text{pkt/sec.}$$

(c). The unknown quantities are the flow rates for the three flow classes,
5/33 and the drop probabilities at each link.




$$x_1 = \frac{\sqrt{2}}{\text{RTT for flows of class 1}\sqrt{\text{drop prob seen by flows of class 1}}} \qquad x_2 = \frac{\sqrt{2}}{\text{RTT for flows of class 2}\sqrt{\text{drop prob seen by flows of class 2}}} \qquad x_3 = \frac{\sqrt{2}}{\text{RTT for flows of class 3}\sqrt{\text{drop prob seen by flows of class 3}}}$$

$$p_A = \frac{(5x_1 + 2x_3 - C_A)^+}{5x_1 + 2x_3} \qquad p_B = \frac{(7x_2 + 2x_3 - C_B)^+}{7x_2 + 2x_3}.$$

[it would also be reasonable to use $x_3(1-p_A)$
[rather than $x_3$ in this equation.

(d). Putting in the constants we're given,
5/33 and calculating the overall drop probabilities for flows of class 3.
[as in the "web cache placement" example

$$x_1 = \frac{\sqrt{2}}{0.01\sqrt{p_A}} \qquad x_2 = \frac{\sqrt{2}}{0.1\sqrt{p_B}} \qquad x_3 = \frac{\sqrt{2}}{0.11\sqrt{1-(1-p_A)(1-p_B)}}$$

$$p_A = \frac{(5x_1 + 2x_3 - 120)^+}{5x_1 + 2x_3} \qquad p_B = \frac{(7x_2 + 2x_3 - 360)^+}{7x_2 + 2x_3}.$$

(e). Pick plausible starting values, e.g.

4/33

$$x_3 = \min\left(\frac{c_A}{7}, \frac{c_B}{9}\right) \approx 18 \text{ pkt/sec.}$$

ie flows of class 3 are bottlenecked at either link A or link B

$$x_1 = \frac{c_A - 2x_3}{5}$$

$$x_2 = \frac{c_B - 2x_3}{7}$$

ie flows of class 1 and 2 mop up the excess capacity at links A and B

$$p_A = 0.1$$

$$p_B = 0.1.$$

don't choose $p_A = 0$ or $p_B = 0$, since otherwise the TCP throughput equation doesn't work.

(f). The fixed point approach is to take these starting values, then iterate the equations from (d) until they stabilize.

8/33

Advanced note. If you're unlucky, you may end up with $p_A = 0$ or $p_B = 0$ ~~at~~ some iteration. This messes up the TCP throughput equation, since $\frac{\sqrt{2}}{RTT\sqrt{0}} = \infty$. This signifies that, if there are no drops ever, TCP will continue taking more and more bandwidth. This can't happen in practice — when TCP's flow rate gets large enough, eventually there will be drops, making $p > 0$. You can get around this by choosing a smoothed function for $p$ instead of $\left(\frac{y-c}{y}\right)^+$.

I don't expect you to spot this or to know what to do, in the exam (but of course I'm delighted to see any interesting comments about the situation).

clarity, style, sense:
5/33.