058020

UNIVERSITY OF YORK

MSc Examinations 2003
MATHEMATICS
Networks II

Time Allowed: 2 hours (plus 15 minutes preparation time).

Answer four questions.
Calculators may not be used.
Candidates are allowed an additional 15 minutes for preparation prior to the start of the examination. Candidates should write on the yellow paper provided during this preparation time. Candidates should only write on the white lined paper after the start of the examination. The examiners will not consider material on the yellow paper.

1. (i) Describe the OSI layers model listing the seven layers and *briefly* describing each. At which level is the IP protocol? At which level is the UDP protocol?

(15 marks)

(ii) Consider the Markov chain with states numbered 1 and 2 and the following transition matrix:

$$\mathbf{P} = \begin{bmatrix} 1/4 & 3/4 \\ 1/2 & 1/2 \end{bmatrix}$$

- Calculate the eigenvalues of $\mathbf{P}$ and hence give the general form for $p_{ij}^{(n)}$ (the n step transition probabilty from state i to state j).
- From inspection of the matrix calculate the values of $p_{11}^{(0)}$ and $p_{11}^{(1)}$.
- Hence calculate $p_{11}^{(n)}$ and check your answer by calculating $p_{11}^{(2)}$ directly from the matrix.
- From your expression for $p_{11}^{(n)}$ (or by any other means) calculate the equilibrium probabilities π_1 and π_2.

(10 marks)

2. (i) Draw a graph showing the arrivals and departures against time τ for a *non FIFO* queue assuming that the system starts empty.

Mark on your graph:

- The arrival time of the ith customer, t_i.
- The total time in the system of the ith customer, $T(i)$.
- The number of arrivals up to time τ, $\alpha(\tau)$.

(5 marks)

(ii) Let $\beta(\tau)$ be the number of departures up to time τ
Let $N(\tau)$ be the number in the system at time τ.
Let N_t be the average value of $N(\tau)$ in the interval $[0, t]$.
Assume that the following limits exist:

$$N = \lim_{t\to\infty} N_t$$

$$\lambda = \lim_{t\to\infty} \frac{\alpha(t)}{t} = \lim_{t\to\infty} \frac{\beta(t)}{t}$$

Prove Little's theorem given these assumptions and the assumption that the queue is initially empty ($N(0) = 0$).

(15 marks)

(iii) Consider a buffer which holds L bytes. The buffer is drained by an ouput line which sends packets at a rate of λ packets per second. The average length of a packet is $\overline{X}$ bytes.

A network engineer decides that the average occupancy of the buffer should be no more than a proportion α of its total size L. Assuming no packets are lost, what is the maximum average queuing delay for packets.

(5 marks)

continued on next page

3. Consider the $M/G/1$ queuing system. Assume that customer service times are independent and identically distributed.

Let X_i be the service time of the ith customer and X be the time series of these service times.

The mean service time is $\overline{X} = E[X] = 1/\mu$ (where μ is the mean service rate).

The second moment of service time is $\overline{X^2} = E[X^2]$.

R_i is the residual service time remaining to the customer being served when the ith customer arrives at the queue. (If the system is empty when the ith customer arrives then $R_i = 0$).

$r(\tau)$ is the residual service time at time τ.

(i) Draw a graph of $r(\tau)$ versus τ. Mark on your graph X_i.
(5 marks)

(ii) Choose a time t, where the system is empty and define $M(t)$ as the number of customers served by time t. Assume that:

- The mean residual time $R = lim_{i\to\infty} E[R_i]$, where $R = \lim_{t\to\infty} \frac{1}{t}\int_0^t r(\tau)d\tau$.
- $\lim_{t\to\infty} \frac{M(t)}{t} = \lambda$ where λ is the mean arrival rate.

From your diagram show that:

$$R = \frac{1}{2}\lambda\overline{X^2}$$

(10 marks)

(iii) Let N_Q be the mean number of customers in the queue. The mean waiting time in the queue, W is given by:

$$W = R + \frac{1}{\mu}N_Q$$

From this and your previous answer, prove the Pollaczek-Khinchin formula:

$$W = \frac{\lambda\overline{X^2}}{2(1-\rho)}$$

where $\rho = \lambda/\mu$.
(2 marks)

(iv) Consider a supermarket where customers arrive as a Poisson process with a rate λ. Two checkouts are available, and both serve customers in a time N/μ where N is the number of items in the customer's trolley. Queue one is a five items or less queue and each customer buys between one and twenty items with equal probabilities (chosen independently). Assuming that λ and μ are such that the Pollaczek-Khinchin formula above applies, show that customers' expected queuing times W_1 and W_2 in queues one and two are given by:

$$W_1 = \frac{\lambda \sum_{i=1}^{5} i^2}{40\mu^2(1 - 3\lambda/4\mu)}$$

$$W_2 = \frac{\lambda \sum_{i=6}^{20} i^2}{40\mu^2(1 - 39\lambda/4\mu)}$$

(You may assume that every customer goes to the queue appropriate for the number of items they have). **(8 marks)**.

4. Consider the $M/M/1/1$ queuing system (the final 1 means that only one customer can be in the system at once — customers arriving to find another customer in service will be turned away from the system). This is to be modelled as a Birth-Death process with the following coefficients:

$$\lambda_k = \begin{cases} \lambda & k = 0 \\ 0 & k = 1 \end{cases}$$

$$\mu_k = \begin{cases} 0 & k = 0 \\ \mu & k = 1 \end{cases}$$

(i) Show that differential-difference equations for $P_0(t)$ and $P_1(t)$ the probabilities that the system is empty or serving a customer at time t are:

$$\frac{dP_0(t)}{dt} = -\lambda P_0(t) + \mu P_1(t)$$

$$\frac{dP_1(t)}{dt} = -\mu P_1(t) + \lambda P_0(t)$$

(5 marks)

(ii) Using the fact that $P_0(t) + P_1(t) = 1$ solve the above to get a general expression for $P_0(t)$.

(12 marks)

(iii) Given the specific value $P_0(0)$ at $t = 0$, show further that the solution is:

$$P_0(t) = \left(P_0(0) - \frac{\mu}{\lambda + \mu}\right) e^{(-\lambda-\mu)t} + \frac{\mu}{\lambda + \mu}$$

(8 marks)

5. (i) Describe Dijkstra's algorithm for finding shortest paths in a weighted digraph $G = (\mathcal{N}, \mathcal{A})$ with arc weights w_{ij} for each arc $(i, j) \in \mathcal{A}$ and with the notational convenience that $w_{ij} = \infty$ if $(i, j) \notin \mathcal{A}$.
(10 marks)

(ii) Prove that Dijkstra's algorithm does find a shortest path.
(10 marks)

(iii) Write down the iterations of Dijkstra's algorithm with the sets of permanent and temporary nodes at each iteration (with costs) for the path from O to D in the network pictured below. Therefore indicate the shortest path through the network and its cost.
(5 marks)

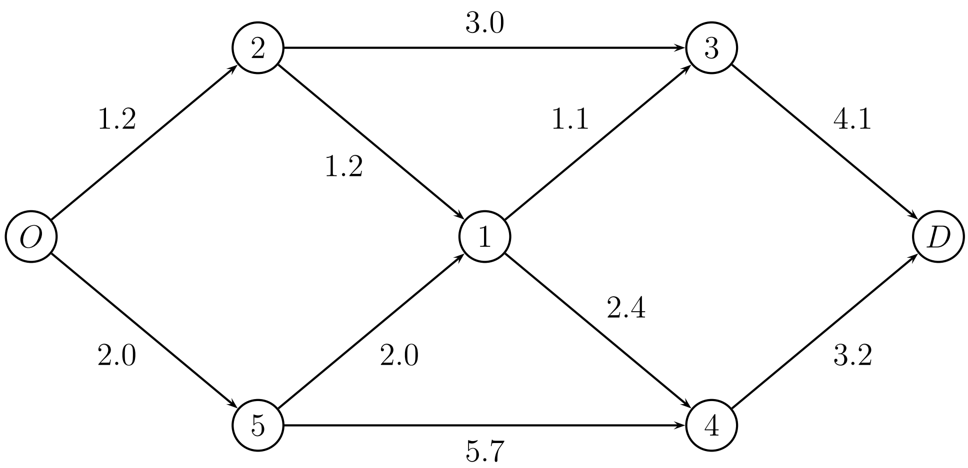

Figure 1: Weighted graph for Dijkstra's algorithm

1. (i) Many more details of the OSI model were presented in lectures — this sample answer just lists the most relevant features.

1: Layer One is the physical layer. This is the hardware which makes up the network.

2: Layer Two is the data link layer (or logical layer). This provides a connection between adjacent (physically or logically) machines in a network.

3: Layer Three is the network layer. This allows data to get between any to machines on a network.

4: Layer Four is the transport layer. This ensures the connection from end-to-end, guaranteeing losslessness if necessary and providing basic flow control.

5: Layer Five is the session layer. This watches over an entire connection.

6: Layer Six is the presentation layer. This takes care of common tasks (such as internationalisation of character sets) which would be inappropriate at other layers.

7: Layer Seven is the application layer. This is the layer where software (web browsers, email etc) connect to the network.

The IP protocol is at layer 3. The UDP protocol is at layer 4

(ii) Using $|\mathbf{P} - \lambda\mathbf{I}| = 0$ we get:
$(1/4 - \lambda)(1/2 - \lambda) - 3/8 = 0$
The quadratic formula gives us $\lambda_1 = 1$ and $\lambda_2 = -1/4$.
This gives us the general form:

$$p_{ij}^{(n)} = A + B(\frac{-1}{4})^n$$

Direct calculation gives us $p_{11}^{(0)} = 1$ and $p_{11}^{(1)} = 1/4$. Subsituting into the above gives: $A + B = 1$ and $A - 1/4B = 1/4$ which solves to the general form:

$$p_{11}^{(n)} = 2/5 + 3/5(\frac{-1}{4})^n$$

From the matrix $p_{11}^{(2)} = 1/16 + 3/8 = 7/16$. This is equal to the value calculated above $p_{11} = 2/5 + 3/5(-1/4)^2 = (2.16 + 3)/(5.16) = 7/16$.

From the above equation $p_{11}^{(n)} \to 2/5$ as $n \to \infty$ and therefore $\pi_1 = 2/5$.

2. (i) Figure 2 shows the situation if we don't assume FIFO.

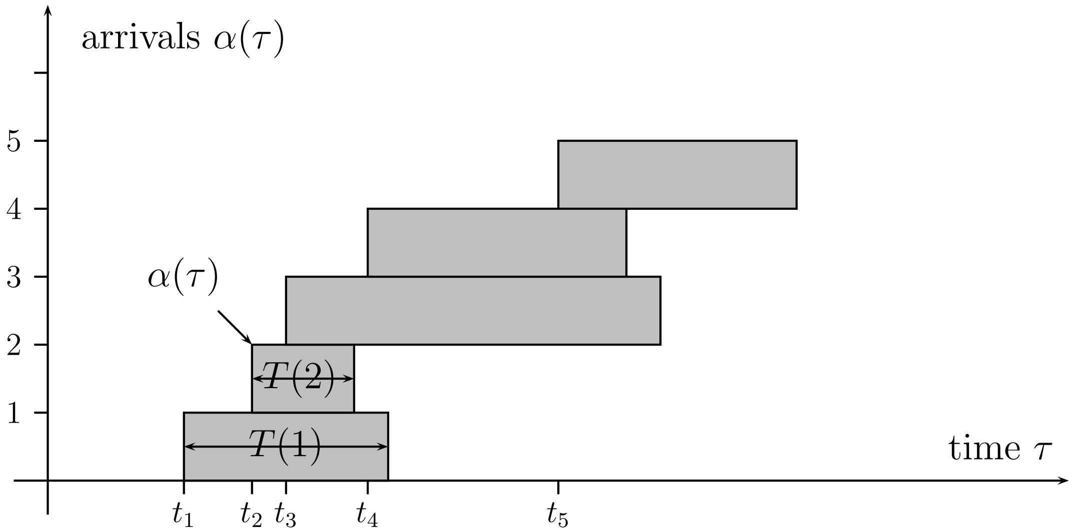

Figure 2: Little's Theorem in a non FIFO System

(ii) If N_t is the mean value of $N(\tau)$ taken over the interval $[0, t]$ then it is clear that:

$$N_t = \frac{1}{t}\int_0^t N(\tau)d\tau \tag{1}$$

Now, it is clear that the shaded area in the interval $[0, t]$ is given by:

$$A(t) = \int_0^t N(\tau)d\tau \tag{2}$$

even though, in the diagram, $N(\tau)$ is not necessarily a continuous vertical slice of the area (consider, for example, the situation at time t_4 in the diagram).

Now, we define the following:

$D(t)$ is the set of customers who have departed the system by time t.
$\overline{D}(t)$ is the set of customers who are still in the system at time t.
The delay experienced up to time t by a customer still in the system at time t is $t - t_i$. Therefore we can say:

$$A(t) = \sum_{i \in D(t)} T_i + \sum_{i \in \overline{D}(t)} (t - t_i) \tag{3}$$

Therefore, equating these and dividing by t we get:

$$\frac{1}{t} \int_0^t N(\tau) d\tau = \frac{1}{t} \sum_{i \in D(t)} T(i) + \frac{1}{t} \sum_{i \in \overline{D}(t)} (t - t_i) \tag{4}$$

Up to time t, the average arrival rate λ_t is given by:

$$\lambda_t = \frac{|D(t) + \overline{D}(t)|}{t} \tag{5}$$

Up to time t, the average waiting time T_t is given by the total waiting time of all the customers so far over the total number of customers entering the system so far (up to time t):

$$T_t = \frac{\sum_{i \in D(t)} T(i) + \sum_{i \in \overline{D}(t)} (t - t_i)}{|D(t) + \overline{D}(t)|} \tag{6}$$

Substituting these two equations and our previous equation (1) into (4) we therefore have:

$$N_t = \lambda_t T_t \tag{7}$$

which, in the limit as $t \to \infty$ gives us Little's Theorem.

$$N = \lambda T \tag{8}$$

(iii) Take Little's Theorem $N = \lambda T$ where T is the average delay for packets in the queue and N is the average length of the queue in packets. Since $N\overline{X} \leq \alpha L$ then we have $\alpha L / \overline{X} \geq \lambda T$ and therefore:

$$T \leq \frac{\alpha L}{\lambda \overline{X}} \tag{9}$$

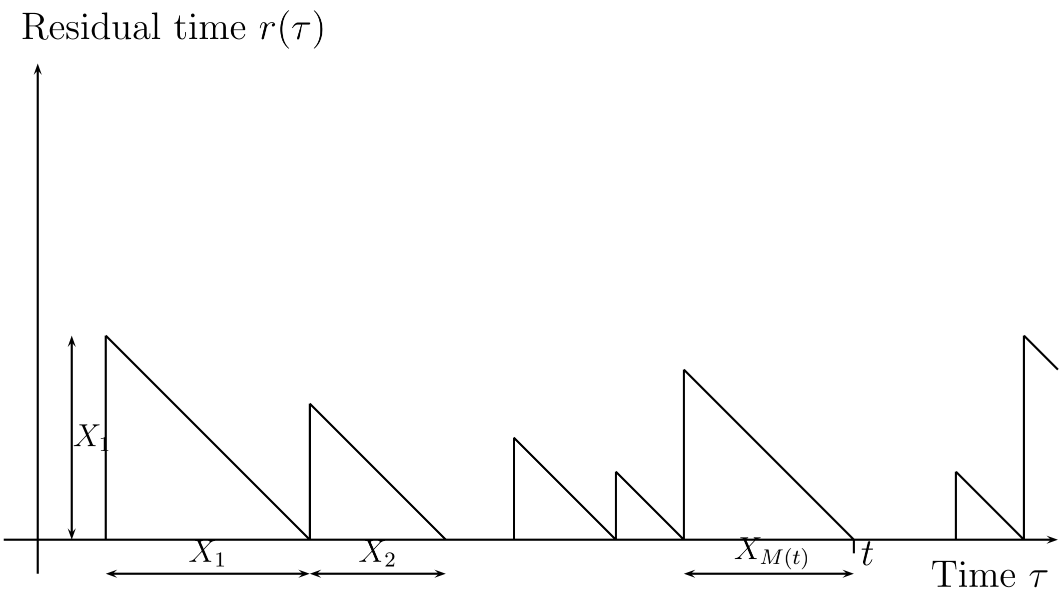

Figure 3: Service Time of Arrivals at an $M/G/1$ queue.

3. (i) Figure 3 shows residual times for the $M/G/1$ queuing system.

(ii) Expressing the area under the graph in two different ways we get:

$$\frac{1}{t}\int_0^t r(\tau)d\tau = \frac{1}{t}\sum_{i=1}^{M(t)} \frac{1}{2}X_i^2$$

Taking limits as $t \to \infty$ we get:

$$R = \lim_{t\to\infty} \frac{M(t)}{t} \frac{\sum_{i=1}^{M(t)} X_i^2}{2M(t)}$$

Now, since the output rate must be the input rate λ we have:

$$\lim_{t\to\infty} \frac{M(t)}{t} = \lambda$$

Substituting above gives us the required result:

$$R = \frac{1}{2}\lambda\overline{X^2}$$

(iii) From Little's Theorem $N_Q = W\lambda$. Therefore:

$$W = \frac{1}{2}\lambda\overline{X^2} + W\rho$$

leading to the required:

$$W = \frac{\lambda\overline{X^2}}{2(1-\rho)}$$

(iv) Since the arrivals choose their queue in an i.i.d. way we have two separate $M/G/1$ queues with $\lambda_1 = \lambda/4$ and $\lambda_2 = 3\lambda/4$. Let $\overline{X_i^2}$ $i = (1, 2)$ be the second moment of service time for customers in queue i. We can say that:

$$\overline{X_1^2} = \frac{\sum_{i=1}^{5} i^2}{5\mu^2}$$

and

$$\overline{X_2^2} = \frac{\sum_{i=6}^{20} i^2}{15\mu^2}$$

The mean service time for customers in queue 1 (call it $1/\mu_1$) is given by:

$$\frac{1}{\mu_1} = \frac{\sum_{i=1}^{5} i/\mu}{5} = 3/\mu$$

and simularly

$$\frac{1}{\mu_2} = \frac{\sum_{i=6}^{20} i/\mu}{15} = 13/\mu$$

Substituting into the P-K equations gives us the required answers.

4. (i)

$$P_0(t+\Delta t) = (1-\lambda\Delta t)P_0(t) + \mu P_1(t)\Delta t + o(\Delta t)$$

$$\frac{P_0(t+\Delta t) - P_0(t)}{\Delta t} = -\lambda P_0(t) + \mu P_1(t) + \frac{o(\Delta t)}{\Delta t}$$

Taking the limit as $t \to 0$ we get:

$$\frac{dP_0(t)}{dt} = -\lambda P_0(t) + \mu P_1(t)$$

the second equation can be trivially obtained by observing that since $P_0(t) + P_1(t) = 1$ then $dP_0(t)/dt = -dP_1(t)/dt$.

(ii) Substitution from $P_0(t) + P_1(t) = 1$ gives us:

$$\frac{dP_0(t)}{dt} = (-\lambda - \mu)P_0(t) + \mu$$

The solution is, by inspection, of the form:

$$P_0(t) = Ke^{(-\lambda-\mu)t} + C$$

Differentiating gives us:

$$\frac{dP_0(t)}{dt} = K(-\lambda - \mu)e^{(-\lambda-\mu)t}$$

comparing with our original equation we get:

$$K(-\lambda - \mu)e^{(-\lambda-\mu)t} = (-\lambda - \mu)(Ke^{(-\lambda-\mu)t)} + C) + \mu$$

Giving us $C = \mu/(\lambda + \mu)$.
This gives us:

$$P_0(t) = Ke^{(-\lambda-\mu)t} + \frac{\mu}{\lambda + \mu}$$

(iii) At $t = 0$ we have:

$$P_0(0) = K + \frac{\mu}{\lambda + \mu}$$

Giving $K = P_0(0) - \frac{\mu}{\lambda+\mu}$ and leading to our final equation:

$$P_0(t) = \left(P_0(0) - \frac{\mu}{\lambda + \mu}\right) e^{(-\lambda-\mu)t} + \frac{\mu}{\lambda + \mu}$$

5. (i) The algorithm finds the shortest path to all nodes from an origin node, on a graph $G = (\mathcal{N}, \mathcal{A})$. It requires that all arc weights are non-negative $\forall (i, j) \in \mathcal{A} : w_{ij} \geq 0$.

Dijkstra's Algorithm involves the labelling of a set of permanent nodes P and node distances D_j to each node $j \in \mathcal{N}$.

Assume that we wish to find the shortest paths from node 1 to all nodes. Then we begin with: $P = \{1\}$ and $D_1 = 0$ and $D_j = w_1 j$ where $j \neq 1$. Dijkstra's algorithm then consists of following the procedure:

(a) Find the next closest node. Find $i \notin P$ such that:

$$D_i = \min_{j \notin P} D_j$$

(b) Update our set of permanently labelled nodes and our nodes distances:
$P := P \cup \{i\}$.

(c) If all nodes in $\mathcal{N}$ are also in P then we have finished so stop here.

(d) Update the temporary (distance) labels for the new node i. For all $j \notin P$

$$D_j := \min[D_j, w_{ij} + D_i]$$

(e) Go to the beginning of the algorithm.

(ii) At the beginning of each iteration of Dijkstra's algorithm then:

(a) $D_i \leq D_j$ for all $i \in P$ and $j \notin P$.

(b) D_j is, for all j, the shortest distance from 1 to j using paths with all nodes (except, possibly j) in P.

If this proposition can be proved then we can see that, when P contains every node in $\mathcal{N}$ then all the D_j are shortest paths by the second part of this proposition. Therefore proving the above proposition is equivalent to proving that Dijkstra's algorithm finds shortest paths.

The proposition is trivially true at the first step since P consists only of the origin point (node 1) and D_j is 0 for $j = 1$, is $w_{ij} \geq 0$ for nodes reachable directly from node 1 and ∞ otherwise.

The first condition is simply shown to be satisfied since it is preserved by the formula:

$$D_j := \min[D_j, w_{ij} + D_i]$$

which is applied to all $j \notin P$ when node i is added to the set P.

We show the second condition by induction. We have established already that it is true at the very start of the algorithm. Let us assume it is true for the beginning of some iteration of the algorithm and show that it must then be true at the beginning of the next iteration.

Let node i be the node we are adding to our set P and let D_k be the label of each node k at the beginning of the step. The second condition must, therefore, hold for node $j = i$ (the new node we have added) by our induction hypothesis. It must also hold for all nodes $j \in P$ by part one of the proposition which is already proven. It remains to prove that the second condition of the proposition is met for $j \notin P \cup \{i\}$.

Consider a path from 1 to j which is shortest amongst those with all nodes except j in $P \cup \{i\}$ and let D'_j be the corresponding shortest distance. Such a path must contain a path from 1 to some node $r \in P \cup \{i\}$ and an arc(r, j). We have already established that the length of the path from 1 to r must be D_r and therefore we have:

$$D'_j = \min_{r \in P \cup \{i\}} [D_r + w_{rj}] = \min[\min_{r \in P} [D_r + w_{rj}], D_i + w_{ij}]$$

However, by our hypothesis $D_j = \min_{r \in P} [D_r + w_{rj}]$ therefore, $D'_j = min[D_j, D_i + w_{ij}]$ which is exactly what is set by the fourth step of the algorithm. Thus, after any iteration of the algorithm, the second part of the propostion is true if it was true at the beginning of the iteration. Thus the proof by induction is complete.

(iii)

Permanent nodes	Temporary Nodes
O (0)	2 (1.2), 5 (2.0)
O(0), 2(1.2)	3(4.2), 1 (2.4), 5(2.0)
O(0), 2(1.2), 5(2.0)	1 (2.4), 3(4.2), 4(7.7)
O(0), 2(1.2), 5(2.0), 1(2.4)	3(3.5), 4(4.8)
O(0), 2(1.2), 5(2.0), 1(2.4), 3(3.5)	4(4.8), D(7.6)
O(0), 2(1.2), 5(2.0), 1(2.4), 3(3.5), 4(4.8)	D(7.6)

Table 1: Nodes for Dijkstra's Algorithm

The final route is $O \to 2 \to 1 \to 3 \to D$.