

**Examination in School of Mathematical Sciences
Semester 2, 2019**

105637	APP MTH 4052	Applied Mathematics Topic F: Complex Network Modelling and Inference
105661	APP MTH 7088	Applied Mathematics Topic F: Complex Network Modelling and Inference

Official Reading Time: 10 mins
Writing Time: 180 mins
Total Duration: 190 mins

NUMBER OF QUESTIONS: 6 TOTAL MARKS: 50

Instructions

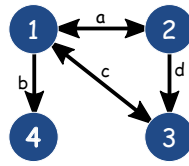
- Attempt all questions.
- Begin each answer on a new page.
- Examination materials must not be removed from the examination room.

Materials

- 1 Blue book is provided.
- Calculators without remote communications facilities are permitted.
- Students are permitted to bring two, double-sided pages of handwritten notes.
- English and foreign-language dictionaries may be used.

DO NOT COMMENCE WRITING UNTIL INSTRUCTED TO DO SO.

1. (a) Graph illustrated below (note edge labels aren't needed but are useful below).



[1 mark]

- (b) The graph is **directed** because A is not symmetric.

[1 mark]

- (c) It is **not** strongly connected. You cannot reach any other node from node 4.

[1 mark]

- (d) Edge list

edge	src	dst
a	1	2
a'	2	1
b	1	4
c	1	3
c'	3	1
d	2	3

Edge labels are not needed, but are useful to match to figure.

[2 marks]

- (e) As the network is directed, we report in- and out-degrees.

[1 mark]

node	in-degree	out-degree
1	2	3
2	1	2
3	2	1
4	1	0

These can be obtained by column- and row-sums of A , respectively.

[1 mark]

- (f) In the context of a directed graph, each edge (or arc) contributes 1 to the total in- and out-degree of the network so

$$|E| = \sum_i k_i^{(in)} = \sum_i k_i^{(out)}.$$

[1 mark]

Here $|E| = 6$ as seen from the table above, and

$$6 = 2 + 1 + 2 + 1 = 3 + 2 + 1 + 0,$$

as required.

[1 mark]

Please turn over for page 3

2. (a) In this context, the path algebra is defined on a set of 2-tuples $S = \{(d, r)\}$ where $d \in \mathbb{R} \cup \infty$ is a distance and $r \in [0, 1]$ is a reliability. Then, define

$$\begin{aligned} (a_d, a_r) \oplus (b_d, b_r) &= \begin{cases} (a_d, a_r), & \text{if } a_d < b_d \text{ or } (a_d = b_d \text{ and } a_r \geq b_r), \\ (b_d, b_r), & \text{if } a_d > b_d \text{ or } (a_d = b_d \text{ and } a_r < b_r), \end{cases} \\ (a_d, a_r) \otimes (b_d, b_r) &= (a_d + b_d, a_r \times b_r), \\ \bar{0} &= (\infty, 0.0), \\ \bar{1} &= (0, 1.0). \end{aligned}$$

[4 marks]

- (b) The weighted adjacency matrix is

$$A = \begin{pmatrix} (0, 1.0) & (2, 0.9) & (1, 0.9) \\ (2, 0.9) & (0, 1.0) & (3, 0.8) \\ (1, 0.9) & (3, 0.8) & (0, 1.0) \end{pmatrix}$$

[1 mark]

- (c) The algorithm for the generalised problem is given below.

Let $D_{ij}^{(k)}$ denote the path capacity from node i to node j using intermediate nodes from 1 to k only.

Initialise: $D_{ij}^{(0)} = d_{ij} \quad \forall i, j \in N$

where d_{ij} is given by the weighted adjacency matrix above.

$V^{(0)} = [0]$, an $|N| \times |N|$ zero matrix.

Step: for $k = 1, 2, \dots, n$, compute new distance estimates

$$D_{ij}^{(k)} = D_{ij}^{(k-1)} \oplus (D_{ik}^{(k-1)} \otimes D_{kj}^{(k-1)}), \quad \forall i \neq j$$

In this case, $\oplus = \max$, and $\otimes = \min$

Compute the predecessor nodes, which in the simple FW algorithm was based on the decision made by $\oplus = \min$ in shortest paths, so here $<$ will be replaced by $>$ to give

If $D_{ij}^{(k)} > D_{ij}^{(k-1)}$ then

$$V_{ij}^{(k)} = k;$$

else

$$V_{ij}^{(k)} = V_{ij}^{(k-1)}$$

Applying this to the problem we get (noting that because the matrix is undirected, we only need to compute the upper triangle)

Please turn over for page 4

Initialisation

$$D_{ij}^{(0)} = \begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline 1 & (0, 1.0) & (2, 0.9) & (1, 0.9) \\ 2 & & (0, 1.0) & (3, 0.8) \\ 3 & & & (0, 1.0) \end{array} \quad V^{(0)} = \begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline 1 & 0 & 0 & 0 \\ 2 & & 0 & 0 \\ 3 & & & 0 \end{array}$$

At each step consider whether the existing $D_{ij}^{(k-1)}$, or the alternative path through node k , *i.e.*, $D_{ik}^{(k-1)} \otimes D_{kj}^{(k-1)}$ is better.

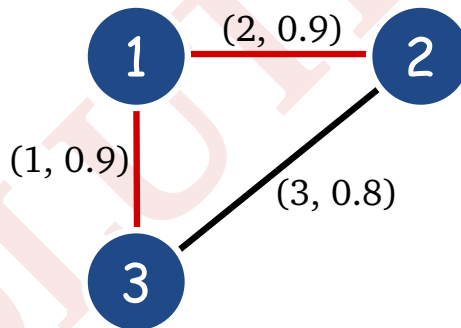
Step 1, $k = 1$, there are two alternative paths from 1-3, the direct path with metric (3, 0.8) and the indirect path with metric (3, 0.81). They have equal length, so the criteria used to choose is the maximum reliability, which is the indirect path, so

$$D_{ij}^{(1)} = \begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline 1 & (0, 1.0) & (2, 0.9) & (1, 0.9) \\ 2 & & (0, 1.0) & (3, 0.81) \\ 3 & & & (0, 1.0) \end{array} \quad V^{(1)} = \begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline 1 & 0 & 0 & 0 \\ 2 & & 0 & 1 \\ 3 & & & 0 \end{array}$$

Step 2, $k = 2$, the alternative paths all have greater length, so there is no change, *i.e.*, $D^{(2)} = D^{(1)}$.

Step 3, $k = 2$, the alternative paths all have greater length, so there is no change, *i.e.*, $D^{(3)} = D^{(2)}$.

Results: best paths between pairs of nodes are show in D in the latest step above. The final path choices derived from V are illustrated in red in the figure below.



[4 marks]

3. (a) When we sample nodes at random, we are filtering the graph, but not changing the underlying model. Nodes are still going to be connected with probability p so we will see a Gilbert-Erdős-Rényi random network $G(n/m, p)$, and thus the observed node degree distribution will be the same as in the original graph but with new parameter λ/m as a result of the thinning, *i.e.*, the node degrees will be distributed as a $\text{Poisson}(\lambda/m)$.

[2 marks]

- (b) Sampling edges at random will result in a biased sample of nodes. Nodes with higher degree are sampled in proportion to their degree. Hence the resulting degree distribution is

$$q_k = \frac{kp_k}{\sum_{k=1}^{\infty} kp_k}.$$

[1 mark]

where p_k is the original degree distribution, $\text{Poisson}(\lambda)$, so

$$p_k = \frac{\lambda^k e^{-\lambda}}{k!}.$$

[1 mark]

Hence using the fact that $\sum_{k=1}^{\infty} kp_k = \lambda$,

$$q_k = \frac{\lambda^k e^{-\lambda}}{(k-1)! \lambda} = \frac{\lambda^{k-1} e^{-\lambda}}{(k-1)!} = p_{k-1}.$$

[1 mark]

- (c) Snowball sampling starts with random “seed” nodes, and then all nodes adjacent to these, and so on out to the radius. In this case, the radius of 1 hop means that snowball sampling will only find nodes one hop from the seed. [1 mark]

Thus we will see the correct node degree for the seed (as all of its adjacent nodes are included in the sampled subgraph). [1 mark]

The Gilbert-Erdős-Rényi random graph in this limit is sparse, and has clustering coefficient approaching zero. Hence, the nodes at one hop distance from the seed will be adjacent to each other with probability approaching zero. Hence these nodes will have degree 1 in the limit. [1 mark]

Thus the final distribution will be a mixture of the original degree distribution, and a probability mass at degree 1. [1 mark]

The weights of the two components will be determined by how many seed nodes vs 1 hop nodes are selected. The number of 1 hop nodes for each seed will be given by the average degree in the network, *i.e.*, $\bar{k} = (n-1)p$. Thus the final probability distribution will be

$$w_k = \begin{cases} \frac{(n-1)p + p_1}{(n-1)p + 1}, & \text{for } k = 1, \\ \frac{p_k}{(n-1)p + 1}, & \text{otherwise.} \end{cases}$$

where p_k is defined above.

[1 mark]

Please turn over for page 6

4. (i) $a^* = 1 \oplus a \oplus a^2 \oplus \dots$ solves the equation because

$$\begin{aligned} (a \otimes a^*) \oplus \bar{1} &= (a \oplus a^2 \oplus a^3 \oplus \dots) \oplus \bar{1} && \text{by distributivity,} \\ &= \bar{1} \oplus a \oplus a^2 \oplus a^3 \oplus \dots && \text{by commutivity of } \oplus, \\ &= a^*. \end{aligned}$$

[3 marks]

(ii) The definition of q -stability is that for all $a \in S$ then $a^{(q)} = a^{(q+1)}$. [1 mark]

By induction $a^{(q)} = a^{(q+k)}$ for all $k = 1, 2, 3, \dots$ (One does not need to (because I didn't in lectures) but could expand on the induction.)

[1 mark]

Now

$$\lim_{k \rightarrow \infty} a^{(q+k)} = a^*,$$

and hence the result.

[1 mark]

5. (a) You could present the pseudo-code (see below), but a more succinct argument is acceptable. For instance:

Imagine that we had a label describing “depth” of a node (given a particular starting point). Then a breadth-first search would search all the nodes at a particular depth before moving onto the next depth. We don’t *a priori* have such a marker, but as we explore the nodes, we assign a depth of one-more-than the current nodes to all neighbours that we haven’t currently explored (and add these nodes to the list of nodes yet to explore). We terminate when all nodes are explored (or we find what we are looking for).

[4 marks]

- (b) A simple search algorithm, which traverses the graph, marks nodes that have been seen, and checks whether we see them again, would suffice (because marking a node twice would indicate the presence of a loop which we cannot have in a tree). For example, using a BFS:

```

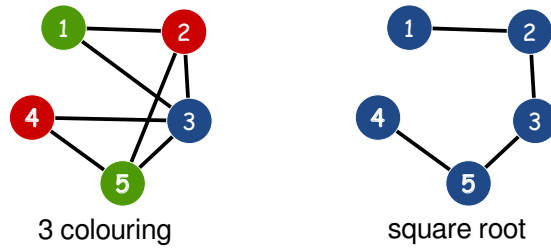
Function TreeTest( $G, i$ );
Input: A Graph  $G = (N, E)$ , and start node  $i \in N$ 
label  $i$  as explored;
create queue  $Q$ ;
put  $i$  on  $Q$ ;
while  $Q$  not empty do
    take  $j$  off the front of  $Q$ ;
    forall  $k \in \text{neighbourhood}\{j\}$  do
        if  $k$  is unexplored then
            label  $k$  as explored;
            put  $k$  on  $Q$ ;
        else
            return FALSE;
        end
    end
end
return TRUE;
    
```

[2 marks]

- (c) Commonly such graphs are used to assemble DNA sequences. [1 mark]
 We would use an algorithm that finds Eulerian cycles on this graph. [1 mark]

6. (a) (i) See figure below:

[1 mark]



(ii) No. Any triangle will require at least 3 colours, e.g., the subgraph formed by nodes 1,2,3.

[1 mark]

(iii) It is the minimum number such that we can find a colouring, so it is 3.

[1 mark]

(iv) Presume we start with node 1, we can choose 3 possible colours. Then there is a choice of 2 possibilities for node 2. Now, the colour of 3 is pre-determined, and thence 5, and thence 4. So in total there are $3 \times 2 = 6$ possibilities.

[1 mark]

(v) The chromatic polynomial takes values given in the following table:

colours	number of colourings
0	0
1	0
2	0
3	6
4	96

The first three terms mean that the polynomial must have factors $t(t - 1)(t - 2)$. When $t = 3$, these result in 6 possibilities, which is the correct factor.

Given a particular value of t , the first three terms show how many possibilities there are for the first three nodes. Then there are $t - 2$ possibilities for node 5, and similarly for node 4. Hence

$$P(G, t) = t(t - 1)(t - 2)^3.$$

[1 mark]

(b) A graph to an integer power, e.g., G^k is the graph containing an edge where-ever there is a no-more-than k hop path between two nodes. The square root graph $H = G^{1/2}$ is a graph such that $H^2 = G$. See figure for the solution here.

[2 marks]

(c) Yes. The edge between 2 and 5 can be drawn so that it doesn't overlap any others, and no others overlap.

[1 mark]

Please turn over for page 9

SOLUTIONS

Please turn over for page 10

NUMBER OF QUESTIONS: 6

MARKS BREAKDOWN:

CORE		24
ADVANCED		26
TOTAL		50

SOLUTIONS