Networks Structure and Dynamics

Université Pierre et Marie Curie
Master d'Informatique spécialité Réseaux

Second session examination

September 2016

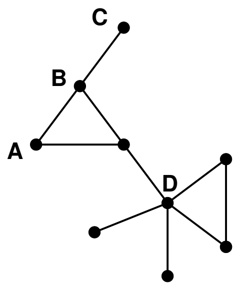
Lionel Tabourier and Fabien Tarissan

The exam is 2 hours long. All documents are allowed. No electronic device is allowed. You must answer the questions on a separate paper. Good luck !

1 Short exercises

Exercise 1 — ***Triangles in a graph.***

Q1. Recall the definition of the clustering coefficient of a node in a graph. What is its value for the nodes A, B, C and D of the following graph ?



Q2. The following algorithm counts the number of triangles nb in a graph. $V(i)$ designates the set of neighbours of node i.
Compute its complexity using the following variables : N, $\delta(i)$ where $\delta(i)$ is the degree of node i. For this question, we suppose that testing if x belongs to a list L demands to go through the entire list. Explain briefly how you obtain this expression.

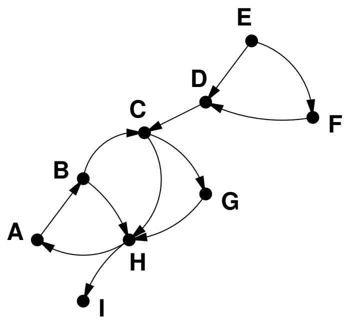
Algorithm 1: Counting the triangles of a graph.

```
nb = 0
for i = 0 to i = N − 1 do
    for j ∈ V(i) do
        for k ∈ V(j) do
            if j > i and k > j and k ∈ V(i) then
                nb++
            end
        end
    end
end
```

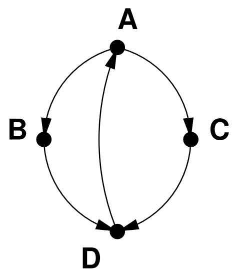
Q3. What does happen without the conditions $j > i$ and $k > j$ of the algorithm ? Does it change the complexity class of the algorithm (justify) ?

Exercise 2 — *PageRank.*

Q1. In the following graph, which are the nodes
a) in the largest strongly connected component ?
b) upstream to the largest strongly connected component ?
c) downstream to the largest strongly connected component ?

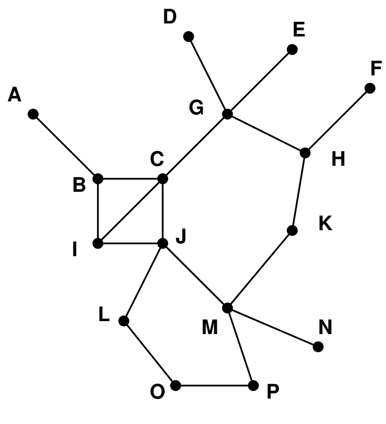


Q2. Give the values of the normalized Pagerank in the following (strongly connected) graph, supposing there is no evaporation process.



Exercise 3 — *Traceroute simulations.*

Considering the following graph :

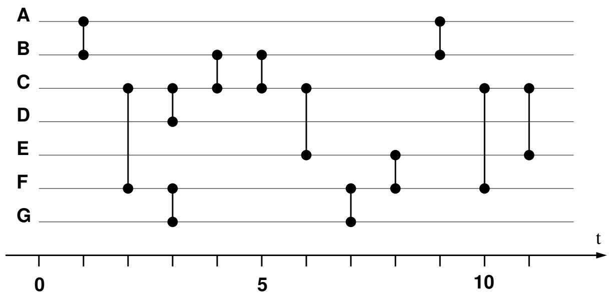


Q1. We compute two Breadth First Search, one from node C and the other from node K. In each case, the result is stored in a table indicating the father of each node. Write a possible result table for each case and draw the corresponding BFS trees.

Q2. Suppose the BFS are shortened to the subset of destinations B, O, D, F. What are the two output tables of the modified BFS ? (we use the same convention as the one seen in the course)

2 Analysis of a dynamical network

The purpose of this exercise is to analyse the dataset represented below. The x-axis represent time and the y-axis the different nodes of the network, we draw a link between two nodes X and Y at t if there is an interaction between X and Y at instant t. To simplify, we consider that a link represents the possibility to exchange information packets at a given moment between two machines (the information transfer is supposed to be instantaneous). We call a hop from X to Y the transfer of a message from X to Y.



Exercise 4 — *Nodes reachability.*

We suppose that A has a packet of information available at instant $t = 0$.

Q1. Can this packet of information emitted by A reach node G before the end of the analysis period ?

Q2. What is the minimum number of hops to reach F following the chronological order ? What is the minimum duration for F to get the message emitted by A ?

Q3. What is the set of nodes that can be reached by packet A ?

Exercise 5 — *Length of the paths.*

We call "length of the path XY" the minimum number of hops to reach Y from X in the dataset.

Q1. Give the set of all possible paths (of minimum length) to reach E from A (between $t = 0$ and the end of the observation period), write them in the following way : A, X, t_1 - X, Y, t_2 - ... What is the corresponding length of the path ?

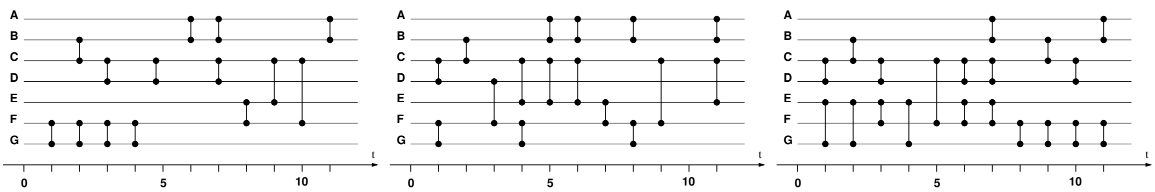
Q2. Give the distribution of the length of the paths from node F (at instant 0) to all other reachable nodes of the network.

Exercise 6 — *Aggregated graph.*

We recall that the aggregated graph is the graph obtained considering all the nodes of a graph and creating a link if there is at least one interaction at a given instant between two nodes.

Q1. Represent the aggregated graph of the dataset proposed.

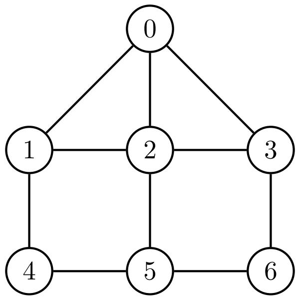
Q2. Among the three following datasets, one as the same aggregated graph as the original dataset, which one ?



Q3. Give the distribution of the length of the paths from node F (at instant 0) to all other reachable nodes of the network for this dataset (following the same principle as in exercise 5). What do you think about it ?

3 Spreading information

In this part, we focus on the behaviour of a *PUSH* protocol to spread information on a network. Let the following structure be the network on which the diffusion occurs :



In the following, we assume that a diffusion trace is composed of a list of pairs of nodes (X, Y) indicating that the node X has sent a message to node Y. In addition, we assume that the trace is chronological and that node 0, source of the observed diffusion, has its own initial copy of the message before the diffusion begins.

Exercise 7 — *Properties of a diffusion trace* Given the following diffusion trace :

0 1
1 2
1 4
2 3
2 5
3 0
3 6
4 5

Answer the following questions :

Q1. Is it possible that the protocol used is a *flooding* protocol ? Justify the answer.

Q2. Compute the *message complexity*, the *infection rate* and the *latency* (max) corresponding the this diffusion trace. For the latency, you will provide also the path leading to your answer.

Q3. What should be the optimum values of these quantities ? Justify briefly your answers.

Q4. Assuming that the diffusion protocol is *probabilistic edge Gossip*, what should be the value of the parameter, given the observed trace ?

Exercise 8 — *Number of copies and diffusion tree* One now wants to refine the metrics used previously by computing two new properties for each node i in the network :

— the number of copies received by i.
— the identifier of the father of i in the diffusion, that is the one that sent the message for the first time to i.

Q1. Given the previous trace, what are the value of these two properties for nodes 0, 2 and 5 of the network.

Given the following (partial) algorithm :

Algorithm 2: Given a diffusion trace, it computes the number of copies and the diffusion tree

```
1  Input : F : file describing the diffusion trace
2           N : number of nodes in the graph
3           S : identifier of the source of the diffusion
4  Output : Two arrays (Copies, Father) where
5           Copies[i] provides the number of copies received by node i
6           Father[i] provides the identifier of the father of i in the diffusion (-1 if it has no father)
7  begin
8      Copies ← Array of N cells
9      Father ← Array of N cells
10
11     for i from 0 to N − 1 do
12         Copies[i] = 0            // no copies at the beginning of the diffusion
13         Father[i] = -1           // no father at the beginning of the diffusion
14     end
15
16     Copies[S]=1
17     Father[S]=S
18
19     while F not empty do
20         (u, v) ← NextLine(F)
21         ...                      // update Copies
22         ...                      // update Father
23         ...
24         ...
25     end
26     Return (Copies, Father)
27 end
```

Q2. Explain line 16 of the algorithm.

Q3. Explain line 17 of the algorithm. Why not use simply the default value -1 used at the beginning of the algorithm (what would be the issue) ?

Q4. Complete the algorithm in order for the arrays Copies and Father to be correctly filled in.