Networks Structure and Dynamics

Université Pierre et Marie Curie
Master d'Informatique spécialité Réseaux

Second session examination - correction

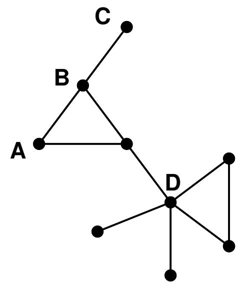
September 2016

Lionel Tabourier and Fabien Tarissan

1 Short exercises

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

Q1. The clustering coefficient of a node x (of degree $\leqslant 2$) is the ratio between its number of connected pairs of neighours and its total number of pairs of neighbours.



On the graph :
— A has 2 neighbours $\Rightarrow$ 1 pair of neighbours, which is connected $\Rightarrow cc(A) = 1$
— B has 3 neighbours $\Rightarrow$ 3 pairs of neighbours, one of them is connected $\Rightarrow cc(B) = 1/3$
— C has 1 neighbour $\Rightarrow cc(C)$ is undefined
— D has 5 neighbours $\Rightarrow$ 4+3+2+1=10 pairs of neighbours, one of them is connected $\Rightarrow cc(D) = 1/10$

Q2. For each node i, we have to go through the whole list of its neighbours. Now, for each node $j \in V(i)$, we have to go through its whole list of neighbours, which is proportional to $\delta(j)$, where $\delta(j)$ is the degree of j. So the complexity is proportional to :

$$\sum_{i=1..N} \sum_{j \in V(i)} \delta(j)$$

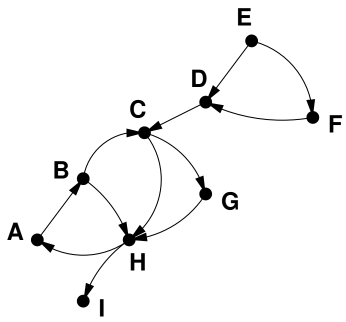
If we make the (rough) approximation that $\delta(j) \simeq \delta(i) \simeq \overline{\delta}$, $\overline{\delta}$ being the average degree, we get that the complexity is proportional to $N\overline{\delta}^2$ (class $\Theta(N\overline{\delta}^2)$)

Q3. Without the conditions $j > i$ and $k > j$, we count the number of triangles several times (6 exactly) as triangles $(i, j, k) = (i, k, j) = (j, i, k) = (j, k, i) = (k, i, j) = (k, j, i)$ are equivalent. The complexity is multiplied by 6, but this doesn't change the complexity class (a quadratic algorithm remains quadratic, if its computation time is multiplied by 6).

Exercise 2 — *PageRank.*

Q1. By definition,

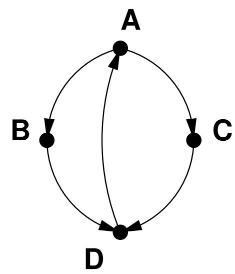
a) The largest strongly connected component is $\{A, B, C, G, H\}$
b) Nodes upstream are $\{E, D, F\}$
c) Nodes downstream are $\{I\}$



Q2. The normalized PageRanks of the nodes are

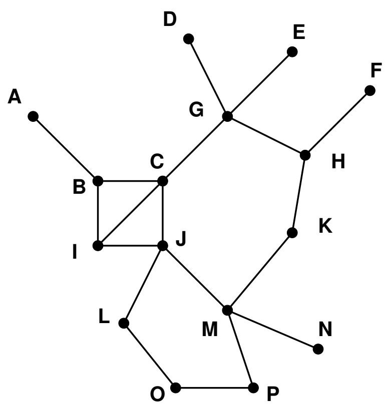
— nodes A and D : 1/3
— nodes B and C : 1/6

It can be found using the fact that if x is the $PR(D)$ in the steady state, then $PR(A) = x$, because A gets all its PageRank from D and A is the only destination of D PageRank. As A PageRank is then splitted equally into B and C PageRank, we must have $PR(C) = PR(B) = x/2$. Finally, the normalization implies that $PR(A) + PR(B) + PR(C) + PR(D) = 1$.

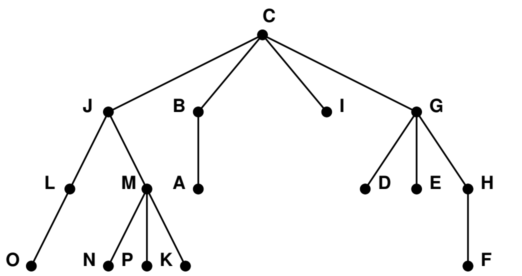


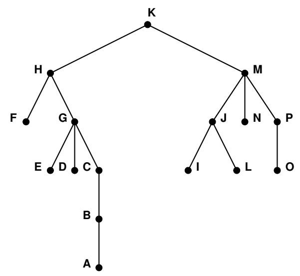
Exercise 3 — *Traceroute simulations.*

Considering the following graph :



Q1. Here are two possible BFS trees obtained respectively from C and K :





And the corresponding tables :

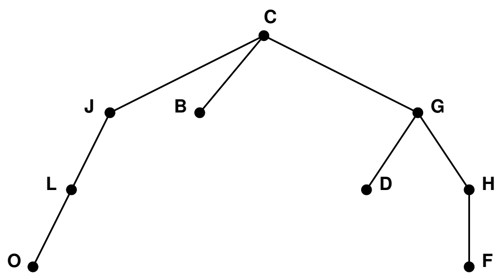
— for C :

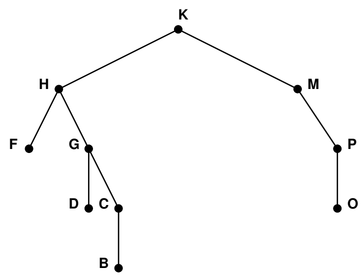
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
B	C	C	G	G	H	C	G	C	C	M	J	J	M	L	M

— for K :

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
B	C	G	G	G	H	H	K	J	M	K	J	K	M	P	M

Q2. We draw below the preceding BFS, shortened to the destinations B, O, D, F :





And the corresponding tables :

— for C :

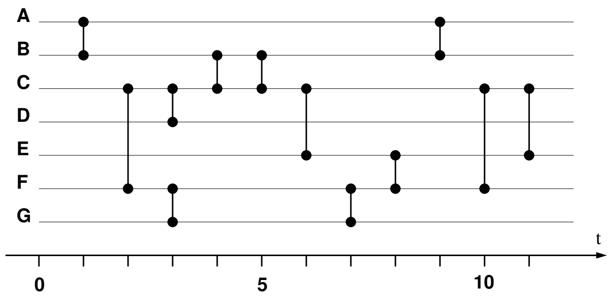
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
-	C	C	G	-	H	C	G	-	C	-	J	-	-	L	-

— for K :

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
-	C	G	G	-	H	H	K	-	-	K	-	K	-	P	M

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. No, the packet of information emitted by A cannot reach node G before the end of the analysis period, because there is no time-respecting path.

Q2. The minimum number of hops to reach F following the chronological order is 3, for example using the path : $A, B, 1 - B, C, 5 - C, F, 10$.

The minimum duration for F to get the message emitted by A is 8, for example using the path : $A, B, 1 - B, C, 5 - C, E, 6 - E, F, 8$.

Q3. The set of nodes that can be reached by packet A is $\{B, C, E, F\}$.

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.

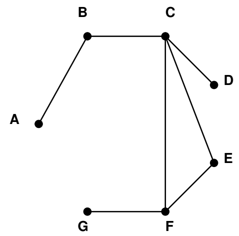
Q1. The possible paths of minimum length = 3 to reach E from A are :
$A, B, 1 - B, C, 4 - C, E, 6$
$A, B, 1 - B, C, 5 - C, E, 6$
$A, B, 1 - B, C, 4 - C, E, 11$
$A, B, 1 - B, C, 5 - C, E, 11$

Q2. Distribution of the length of the paths from node F (at instant 0) to all other reachable nodes :

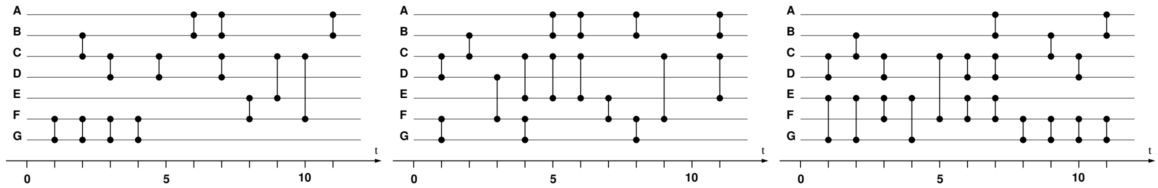
— length 1 : 3 (to C, G and E)
— length 2 : 2 (to B and D)
— length 3 : 1 (to A).

Exercise 6 — ***Aggregated graph.***

Q1. The aggregated graph of the dataset proposed is :



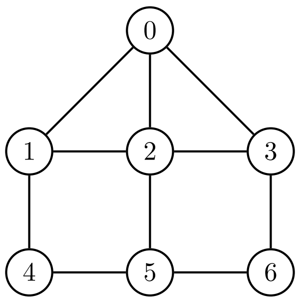
Q2. Among the three following datasets, the one on the left has the same aggregated graph as the dataset.



Q3. Distribution of the length of the paths from node F (at instant 0) to all other reachable nodes :
— length 1 : 3 (to C, G and E) and that's all . . .
So, the aggregated graph contains much less information than the temporal graph, as we can get the same aggregated graph with two different datasets which have very different distributions of their length of paths, and consequently, very different dynamic.

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*** Distribution of contacts : see exam question.

Q1. This protocol cannot be a *flooding*, otherwise there would be a packet sent from 0 to 2.

Q2. — *message complexity* : $M = 9/6$
— *infection rate* : $I = 1$
— *maximum latency* : $L = 4$ because of the spreading from 0 to 6 : $0 - 1 - 2 - 3 - 6$

Q3. Optimal values :
- *message complexity* : $M = 1$
- *infection rate* : $I = 1$
- *maximum latency* : $L = 2$ (max distance from any node to any node reached in the graph)

Q4. Remember that each link can be tested several times. Overall, there are 8 effective deliveries and 20 tests (for example, when a message reach 4, 4 tries to send to its neighbours 1 and 5, so it is two additional tests). So $p = 8/20 = 0.4$.

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.
- 0 : 2 copies, father : itself.
- 2 : 1 copy, father : 1.
- 5 : 1 copies, father : 2.

Algorithm 1: Given a diffusion trace, it computes the number of copies and the diffusion tree

```
Input : F : file describing the diffusion trace
        N : number of nodes in the graph
        S : identifier of the source of the diffusion
Output : Two arrays (Copies, Father) where
        Copies[i] provides the number of copies received by node i
        Father[i] provides the identifier of the father of i in the diffusion (-1 if it has no father)
begin
    Copies ← Array of N cells
    Father ← Array of N cells

    for i from 0 to N − 1 do
        Copies[i] = 0          // no copies at the beginning of the diffusion
        Father[i] = -1         // no father at the beginning of the diffusion
    end

    Copies[S]=1
    Father[S]=S

    while F not empty do
        (u, v) ← NextLine(F)
        Copies[v]++            // update Copies
        if Father[v]==-1       // update Father
        then Father[v] <- u
    end
    Return (Copies, Father)
end
```

Q2. The source is considered as having the original copy before the spreading process starts.

Q3. By convention, the source of the diffusion process is its own father. If we had kept the value -1, either there would be a confusion with a node which is not reached, or a node that sends the message to the source during the process would be considered as its father.

Q4. See algorithm.