

STRUCTURE ET DYNAMIQUE DES RÉSEAUX

Master RES – M2 – Université Pierre et Marie Curie

TP 6

Clémence Magnien, Lionel Tabourier, Fabien Tarissan

`clemence.magnien@lip6.fr`

`lionel.tabourier@lip6.fr`

`fabien.tarissan@lip6.fr`

Le but de cette séance de TP est d'étudier le comportement de différents protocoles de dissémination d'informations dans des réseaux statiques, en analysant leur efficacité en fonction de la topologie sur laquelle a lieu la diffusion. On s'appuiera notamment sur les modèles d'Erdős Rényi et d'Albert-Barabási vus dans les TP précédents.

1 Préliminaires

On supposera dans la suite qu'une trace de diffusion d'information est constituée d'une liste de paires de nœuds (A, B) indiquant chacune que A a diffusé une information à B . Ainsi, la trace suivante indique que le nœud 1 a diffusé une information à 2 et à 3 et que 2 a diffusé cette même information à 3 :

```
1 2
1 3
2 3
```

On rappelle ci-dessous quelques métriques de base liées à l'analyse de l'efficacité d'un protocole de diffusion :

Complexité en messages (C) : traduit à quel point les messages ont été reçus plusieurs fois par les nœuds.

$$C = \frac{\text{Nb Copies}}{\text{Nb nœuds} - 1}$$

Infection (I) : proportion de nœuds qui ont reçus le message

$$I = \frac{\text{Nb nœuds infectés}}{\text{Nb nœuds}}$$

Fiabilité (F) : proportion de messages qui sont reçus par tous les nœuds

$$F = \frac{\text{Nb diffusions complètes}}{\text{Nb diffusions}}$$

Latence max (L) : délai maximal (en nombre de sauts) entre la source et les nœuds destinataires

$$L = \max(\min(\text{source} \rightarrow \text{cible}))$$

où $\text{source} \rightarrow \text{nœuds}$ désigne un chemin dans la trace de diffusion entre le nœud source et le nœud cible .

2 Analyse de traces de diffusion

Exercice 1 (Complexité et infection) *Écrire un programme qui, étant donné une trace de diffusion et le nombre de nœuds du réseau, calcule la complexité en messages ainsi que le taux d'infection lié à cette diffusion.*

Exercice 2 (Fiabilité) *Écrire un programme qui, étant donné une trace de diffusion et le nombre de nœuds du réseau, décide si tous les nœuds du réseaux ont reçu le message.*

En supposant que chaque trace de diffusion est décrite dans un fichier séparé, en déduire un programme qui, étant données plusieurs traces de diffusion, calcule la fiabilité associée à ces traces.

Exercice 3 (Latence max) *Écrire un programme qui, étant donné une trace de diffusion et l'identifiant du nœud source, calcule la latence max associée à cette diffusion.*

3 Inondation

Exercice 4 (Protocole d'inondation) *Écrire un programme qui, étant donné un graphe (décrit dans le format que vous voulez), simule une diffusion appliquant le protocole d'inondation. On prendra soin de choisir aléatoirement la source initiale de la diffusion. Le format de sortie sera celui proposé en section 1.*

Exercice 5 (Simulation sur un graphe aléatoire) *À l'aide du programme précédent, réaliser plusieurs simulations d'inondation sur des graphes aléatoires générés par le modèle d'Erdős Rényi. Analyser l'efficacité de ce protocole sur les graphes aléatoires à l'aide des différentes métriques implémentées précédemment.*

Exercice 6 (Impact du degré moyen) *Étudier l'impact du degré moyen sur l'efficacité du protocole.*

Exercice 7 (Simulation sur un réseau réel) *Reproduire cette simulation puis l'analyse de la trace associée sur le réseau de connexions entre routeurs de l'Internet¹ (que nous nommerons INET) disponible dans le répertoire `/Enseignants-Public/tarissan/sdr/2014/`.*

Comparer l'efficacité observée avec celle obtenue sur les graphes aléatoires (en choisissant notamment un graphe aléatoire de même degré moyen).

1. récupéré à partir du site <http://svnet.u-strasbg.fr/merlin>

Exercice 8 (Simulation sur un graphe aléatoire Albert-Barabási) *Reproduire la même méthodologie en effectuant des simulations de ce protocole de diffusion sur des graphes aléatoires générés par le modèle d'Albert-Barabási.*

4 Algorithmes *Gossip*

On rappelle les différentes variantes de protocole de type *Gossip* :

GossipFF (fixed fanout) : lorsqu'un nœud reçoit un message, il le diffuse auprès d'exactly $fanout$ voisins (autres que l'émetteur), $fanout$ étant un paramètre.

GossipPE (probabilistic edge) : lorsqu'un nœud reçoit un message, il le diffuse auprès de chacun de ses voisins (autres que l'émetteur) avec une probabilité p_e , p_e étant un paramètre.

GossipPB (probabilistic broadcast) : lorsqu'un nœud reçoit un message, il le diffuse auprès de l'ensemble de ses voisins (autres que l'émetteur) avec une probabilité p_b , p_b étant un paramètre.

GossipDT (degree threshold) : lorsqu'un nœud reçoit un message, il le diffuse auprès de tous ses voisins (autres que l'émetteur) si son degré est supérieur ou égal à d , d étant un paramètre.

Exercice 9 (Protocole *GossipFF*) *Écrire un programme qui, étant donné un graphe et une valeur de fanout fixée, simule une diffusion appliquant le protocole GossipFF.*

Exercice 10 (Analyses du protocole *GossipFF*) *À l'aide du programme précédent, réaliser plusieurs simulations du protocole GossipFF sur des graphes aléatoires générés par le modèle d'Erdős Rényi. Analyser l'efficacité de ce protocole. Faire de même avec le réseau INET.*

Exercice 11 (Protocole *GossipPE*) *Écrire un programme qui, étant donné un graphe et une valeur de p_e fixée, simule une diffusion appliquant le protocole GossipPE. Analyser ensuite son efficacité sur des graphes aléatoires ainsi que sur le réseau INET.*

Exercice 12 (Protocole *GossipPB*) *Écrire un programme qui, étant donné un graphe et une valeur de p_b fixée, simule une diffusion appliquant le protocole GossipPB. Analyser ensuite son efficacité sur des graphes aléatoires ainsi que sur le réseau INET.*

Exercice 13 (Protocole *GossipDT*) *Écrire un programme qui, étant donné un graphe et une valeur de d fixée, simule une diffusion appliquant le protocole GossipDT. Analyser ensuite son efficacité sur des graphes aléatoires ainsi que sur le réseau INET.*

Exercice 14 (Albert-Barabási) *Enrichir les analyses précédentes en simulant les différents protocoles sur des graphes générés par le modèle d'Albert-Barabási.*