Option Informatique en Spé MP et MP*

Devoir à rendre après les vacances de Noël

Le protocole d'informations de routage

Résumé

Dans un réseau informatique, le *routage* est l'opération qui consiste à chercher un chemin pour transporter des données d'une source s vers une destination t. Dans ce sujet, nous donnerons une description simplifiée du RIP (*Routing Information Protocol*) qui a été le premier protocole d'information de routage mis en œuvre dans l'Internet et est décrit dans la *RFC 1058*.

Les parties 1, 2 et 3 nous fourniront les bases de théorie des graphes nécessaires.

La partie 4 décrit sommairement le RIP ; on prouve qu'en l'absence de pannes, ce protocole distribué converge. Cette partie est aussi l'occasion de programmer un peu !

La partie 5 étudie le comportement du RIP face aux pannes, ainsi que certains remèdes envisageables.

$\boxed{\textit{Veuillez rédiger chaque partie sur une copie séparée.}}$

Table des matières

1 Graphes : généralités

▶ Soit S un ensemble dont les éléments seront appelés *sommets*. Une *arête* est une paire $a = \{s, t\}$ d'éléments de S ; s et t sont les *extrémités* de a. Nous dirons que a est *incidente* à s et à t. Nous dirons également que s et t sont *voisins* lorsqu'ils sont reliés par une arête, et nous noterons ceci $s \leftrightarrow t$.

▶ Soit A un ensemble d'arêtes ; le couple (S, A) est un *graphe non orienté*. Nous dirons que G est *fini* lorsque S l'est ; dans ce cas, A est lui aussi fini.

▶ L'ossature d'un réseau informatique peut être modélisée par un graphe : les sommets sont des *routeurs* (ordinateurs spécialisés dans l'acheminement des données) et les arêtes sont des liens (filaires, optiques, hertziennes) entre ces routeurs. Notons que cette modélisation écarte volontairement les *hôtes*, c'est-à-dire les ordinateurs des utilisateurs.

▶ Le *degré* d'un sommet s est le nombre d'arêtes incidentes à s ; nous le noterons $\deg(s)$, c'est aussi le nombre des voisins de s. Remarquons que $\deg(s)$ est un entier compris entre 0 (sommet isolé) et $|S| - 1$ (sommet relié à tous les autres).

▶ Un *chemin* de longueur $n \geqslant 0$ menant d'un sommet s à un sommet t est une suite $(x_i)_{0 \leqslant i \leqslant n}$ de sommets tels que $x_0 = s$, $x_n = t$ et $x_{i-1} \leftrightarrow x_i$ pour tout $i \in [\![1, n]\!]$. s et t sont les *extrémités* du chemin. Un chemin $(x_i)_{0 \leqslant i \leqslant n}$ est *élémentaire* si les sommets par lesquels il passe sont tous distincts, à l'exception éventuelle de x_0 et x_n.

▶ Un *cycle* est un chemin de longueur non nulle, dont les extrémités sont confondues. Nous dirons que le cycle est *élémentaire* lorsque le chemin sous-jacent est élémentaire et passe par au moins trois sommets distincts. Vous noterez que (s, t, s) n'est pas un cycle élémentaire.

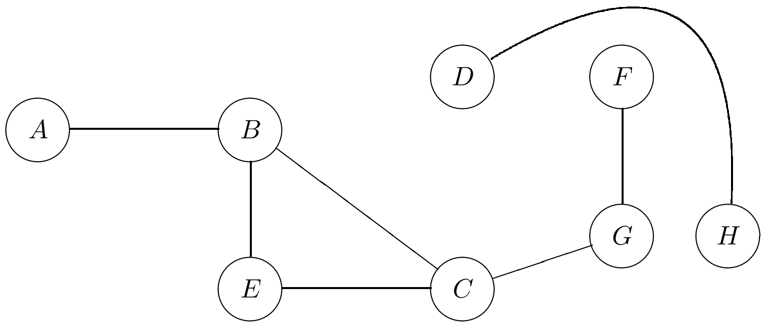

Figure 1: un exemple de graphe

▶ Le graphe de la figure 1 comporte huit sommets et sept arêtes. Le degré du sommet B est 3. On remarque un cycle de longueur 3, passant par B, C et E.

2 Graphes : connectivité

▶ Soient s et t deux sommets. Notons $s \overset{*}{\leftrightarrow} t$ lorsque s et t sont reliés par un chemin.

Question 1 • Montrez que la relation $\overset{*}{\leftrightarrow}$ est une équivalence sur S.

Question 2 • Montrez que si deux sommets sont reliés par un chemin, alors ils sont reliés par un chemin élémentaire. Ce chemin est-il unique ?

▶ Une *composante connexe* de G est une classe modulo $\overset{*}{\leftrightarrow}$. Un graphe est connexe s'il ne compte qu'une composante connexe ; ceci revient à dire que deux sommets quelconques sont toujours reliés par au moins un chemin.

▶ Définissons la *distance* $d(s, t)$ entre les deux sommets s et t d'un graphe G. Si $s \overset{*}{\leftrightarrow} t$, alors $d(s, t)$ la longueur minimale d'un chemin reliant s et t ; sinon, $d(s, t) = +\infty$.

Question 3 • Montrez que d est effectivement une distance.

Question 4 • Soient s et t deux sommets reliés par un chemin. Montrez qu'il existe un chemin élémentaire de longueur $d(s, t)$ menant de s à t.

▶ Le *diamètre* d'un graphe G est la distance maximale entre deux de ses sommets. Ainsi, un graphe est connexe ssi son diamètre est fini.

▶ Le graphe de la figure 1 comporte deux composantes conexes, de diamètres respectifs 4 et 1.

▶ Dans les six questions qui suivent, nous fixons $n \in \mathbb{N}^*$ et nous définissons un graphe $\Gamma_n = (S, A)$ comme suit : S est l'ensemble des permutations de $[\![1, n]\!]$; deux permutations f et g sont voisines ssi il existe une transposition τ telle que $f = \tau \circ g$.

Question 5 • Montrez que l'on définit bien ainsi un graphe non orienté.

Question 6 • Dessiner Γ_3.

Question 7 • Combien Γ_n compte-t-il de sommets ?

Question 8 • Quel est le degré d'un sommet de Γ_n ?

Question 9 • Combien Γ_n compte-t-il d'arêtes ?

Question 10 • Quel est le diamètre de Γ_n ?

▶ Soit s un sommet d'un graphe $G = (S, A)$. Définissons deux suites $(V_n)_{n\in\mathbb{N}}$ et $(W_n)_{n\in\mathbb{N}}$ de parties de A comme suit :

- $V_0 = W_0 = \{s\}$;
- V_{n+1} est l'ensemble des voisins des éléments de V_n qui n'appartiennent pas à W_n ; $W_{n+1} = W_n \cup V_{n+1}$.

Il est clair que la suite $(W_n)_{n\in\mathbb{N}}$ est croissante.

Question 11 • Montrez que $t \in V_n$ ssi $d(s, t) = n$ et $t \in W_n$ ssi $d(s, t) \leqslant n$.

Question 12 • Montrez que $\displaystyle\bigcup_{n\in\mathbb{N}} V_n = \bigcup_{n\in\mathbb{N}} W_n$. Nous noterons $\mathcal{C}(s)$ cette partie de S.

Question 13 • Montrez que $\mathcal{C}(s)$ est la composante connexe de s.

Question 14 • Montrez que si $W_p = W_{p+1}$, alors la suite $(W_n)_{n\geqslant p}$ est constante.

Question 15 • Déduisez des questions précédentes un algorithme de calcul de la composante connexe de s, dans le cas où G est un graphe fini.

Question 16 • Un graphe fini G possède exactement deux sommets de degré impair. Montrez qu'ils sont reliés par un chemin.

3 Graphes : biconnectivité

▶ Dans un réseau informatique, les routeurs et les liens sont sujets à des pannes ou des incidents : chalutier qui accroche un câble sous-marin, météorite qui percute un satellite ... La topologie du réseau doit donc le protéger contre ces incidents. En particulier, la connexité n'est pas suffisante.

▶ Dans la suite, tous les graphes considérés comportent au moins trois sommets. En termes de réseaux informatiques, ce n'est pas vraiment restrictif ...

▶ Soient $G = (S, A)$ un graphe, s un sommet et a une arête. Nous noterons G_a le graphe déduit de G par suppression de l'arête a, et G_s le graphe déduit de G par suppression du sommet s et de toutes les arêtes incidentes à s.

▶ Nous dirons que s est un *point d'articulation* du graphe connexe G si G_s n'est pas connexe. Par exemple, dans la ≪grande≫ composante connexe du graphe présenté à la figure 1, les points d'articulation sont les sommets B, C et G.

▶ Nous dirons que a est un *pont* si G_a n'est pas connexe. Prenons toujours comme exemple le graphe de la figure 1 : l'arête (C, G) est un pont, l'arête (B, E) n'en est pas un.

Question 17 • Montrez que a est un pont ssi aucun cycle élémentaire ne passe par a.

Question 18 • L'étudiant Jean-Marcel Malencontreux affirme que l'une au moins des extrémités d'un pont d'un graphe G est un point d'articulation de G. Que pensez-vous de son affirmation ?

▶ Deux chemins $(x_i)_{0\leqslant i\leqslant p}$ et $(y_j)_{0\leqslant j\leqslant q}$ reliant les sommets s et t sont *disjoints* s'ils n'ont aucun sommet en commun, à part s et t bien entendu.

▶ Un graphe connexe G est *biconnecté* si deux sommets quelconques de G sont toujours reliés par au moins deux chemins disjoints.

Question 19 • Montrez le théorème suivant, dû à Whitney : un graphe G est biconnecté ssi il ne possède pas de point d'articulation.

Question 20 • Montrez qu'un graphe G est biconnecté ssi, quels que soient les sommets s et t distincts de G, il existe un cycle élémentaire qui passe par s et t.

4 Le protocole d'information de routage (RIP)

▶ Nous allons maintenant décrire (sommairement) et étudier le RIP. Un réseau informatique est un graphe fini, dont les sommets sont des *routeurs* baptisés A, B ... et dont les arêtes sont des liens numérotés consécutivement à partir de 1. La figure 2 donne un exemple de réseau, comportant cinq routeurs et six liens.

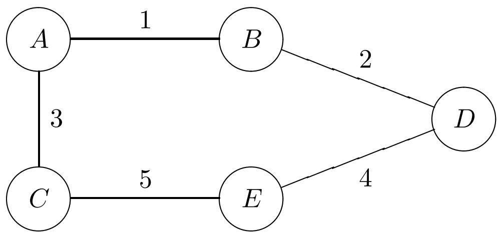

Figure 2: un exemple de réseau informatique

▶ Chaque routeur R tient à jour une *table de routage*, qui résume l'information qu'il détient sur le réseau. Cette table est constituée d'entrées, chacune contenant trois champs :

- le nom d'un routeur R' ;
- le numéro d'un lien à travers lequel R' est accessible ;
- la distance qui sépare R' de R.

▶ Nous comprendrons mieux l'organisation de ces tables en observant comment elles évoluent au cours du temps. Initialement, les routeurs sont tous éteints ; ils vont être allumés l'un après l'autre. Un routeur qui vient d'être allumé ne dispose que de la liste des numéros des liens qui lui sont connectés. La table de routage de A, juste après l'allumage, a l'allure suivante :

de A vers	via	distance
A	*	0

L'unique ligne de la table indique simplement que A est à la distance 0 de lui-même. A diffuse immédiatement cette table à ses voisins B et C (via les liens 1 et 3 respectivement), mais comme ni B ni C n'est allumé, ces informations se perdent dans l'éther. Remarquons que cette table signifie en fait ≪Coucou, je suis un petit nouveau, mon nom est A, je n'ai personne à qui parler pour l'instant mais j'attends de vos nouvelles≫.

▶ Un peu plus tard, B est allumé ; il crée sa table de routage :

de B vers	via	distance
B	*	0

et la diffuse aussitôt à ses voisins A et D (via les liens 1 et 2 respectivement). Seul A est allumé et en prend donc connaissance. Il apprend ainsi que B est le nom du routeur qui est à l'autre bout du lien 1 ; A ajoute cette information à sa table, qui devient :

de A vers	via	distance
A	*	0
B	1	1

Sa table ayant été modifiée, A la diffuse sans délai à ses voisins, afin de les faire profiter des dernières nouvelles. À nouveau, seul B est informé ; il apprend (avec plaisir) l'existence de son voisin A, et sa table devient

de B vers	via	distance
B	*	0
A	1	1

B procède ensuite à la diffusion de sa table mise à jour, tout comme vient de le faire A : à nouveau, seul A en prend connaissance, mais cette fois il n'apprend rien de neuf. Constatons que le réseau est dans un état stable : si nous décidions de n'allumer aucun autre routeur, et si aucune panne ne se produsisait, les choses en resteraient là !

▶ Allumons D maintenant ; il crée sa table :

de D vers	via	distance
D	*	0

puis la diffuse à ses voisins B et E (via les liens 2 et 4 respectivement). À réception, B met à jour sa table, qui devient :

de B vers	via	distance
B	*	0
A	1	1
D	2	1

Comme sa table a été modifiée, B la diffuse aussi sec à ses voisins A et D, dont les tables deviennent respectivement :

de A vers	via	distance
A	*	0
B	1	1
D	1	2

de D vers	via	distance
D	*	0
B	2	1
A	2	2

Expliquons brièvement : lorsque A reçoit la table de routage de B, il apprend l'existence d'un nouveau routeur (D) que B affirme savoir atteindre par un chemin de longueur 1 ; A en déduit que lui-même peut atteindre D par un chemin de longueur 2, qui commence par le lien 1 ; pour la suite du chemin, il se repose sur B. La situation est parfaitement symétrique pour D.

Question 21 • A et D viennent de modifier leurs tables de routage respectives et vont donc les réémettre : montrer que ceci ne changera pas l'état du réseau.

Question 22 • Allumons le routeur E : décrivez ce qui se passe, et constatez que le réseau converge vers un état stable. Remarque : vous devrez choisir l'ordre dans lequel les tables sont diffusées.

Question 23 • Déterminez l'état vers lequel converge le réseau après allumage du dernier routeur (C).

▶ Revenons sur la signification des informations contenues dans chaque table de routage : si par exemple la table de A contient l'entrée $(E, 1, 3)$ celle-ci doit être interprétée comme suit : si A veut transmettre un paquet à E, il peut l'expédier via le lien numéro 1 (ce qui revient à le confier au routeur B, mais à la limite A se moque bien de savoir qui est réellement à l'autre extrémité du lien), et que le paquet traversera 3 liens.

Question 24 • Montrez que, quel que soit l'ordre dans lequel on allume les routeurs, un tel réseau converge vers un état stable $\mathcal{S}$, et ce quel que soit l'ordre de diffusion des mises à jour, sous les hypothèses suivantes :

- le réseau achemine toute mise à jour en un temps fini ;
- chaque routeur communique les modifications dont il a connaissance, avec un délai fini.

Question 25 • Cet état stable est-il unique ?

Question 26 • Soient s et t deux routeurs du réseau ; montrez que s'il existe dans l'état $\mathcal{S}$ un chemin menant de s à t, alors les tables de routage permettent *effectivement* de trouver un tel chemin.

Question 27 • Montrez que ce chemin est unique, et que sa longueur est minimale.

Question 28 • Au vu de l'exemple présenté à la figure 2, on pourrait croire qu'une autorité supérieure doit être chargée de numéroter les liens. Qu'en pensez-vous ?

▶ Soit (t, n, t') un triplet où n est le numéro d'un lien, tandis que t et t' sont les tables de routage des deux routeurs ρ et ρ' situés aux extrémités de ce lien. Notons alors $\mu(t, n, t')$ la nouvelle table de ρ, mise à jour après réception par celui-ci de la table t' via le lien n : μ est la fonction de mise à jour d'une table de routage.

▶ Nous utiliserons les types Caml suivants :

```
type dist = Fini of int | Infini ;;
type route = { cible : string ; via : int ; distance : dist } ;;
```

Le type `dist` permet de représenter une distance (finie ou infinie). Une table de routage sera représentée par une `route list` ; à titre d'exemple, voici la table de routage de A, une fois l'état stable atteint après allumage des routeurs A, B et D :

```
let t = [
 {cible = "A" ; via = 0 ; dist = Fini 0 } ;
 {cible = "B" ; via = 1 ; dist = Fini 1 } ;
 {cible = "D" ; via = 1 ; dist = Fini 2 } ;
] ;;
```

Question 29 • Rédigez en Caml une fonction de signature

```
min_dist : dist -> dist -> dist
```

spécifiée comme suit : `min_dist d d'` calcule la plus petite des deux distances d et d' ; la relation d'ordre sur $\mathbb{N} \cup \{+\infty\}$ est définie de façon naturelle.

Question 30 • Rédigez en Caml une fonction de signature

```
mise_à_jour_route : route list -> int -> route -> route list
```

spécifiée comme suit : `mise_à_jour t n r` construit la table $\mu(t, n, t')$, où t' est la table réduite à la seule route r.

Question 31 • Rédigez en Caml une fonction de signature

```
mise_à_jour : route list -> int -> route list -> route list
```

spécifiée comme suit : `mise_à_jour t n t'` construit la table $\mu(t, n, t')$.

5 RIP face aux pannes

▶ Des incidents peuvent se produire : coupure d'un lien, panne d'un routeur. Pour détecter ces incidents, le protocole RIP impose deux règles supplémentaires :

- chaque routeur diffuse sa table à ses voisins immédiats, toutes les trente secondes ;
- si un routeur ne reçoit aucune information via un lien donné pendant une période de 180 secondes, il déclare que ce lien est ≪mort≫ : dans sa table, il modifie toute entrée dont le champ *via* indique ce lien, en donnant la valeur $+\infty$ au champ *distance*.

Question 32 • Supposons que le réseau comporte un millier de routeurs, et que les liens aient un débit de 10 Mbits par seconde. Quel pourcentage de la bande passante est consommé par la diffusion des tables de routage ? Vous estimerez à vingt octets la taille de chaque entrée dans une table de routage.

Question 33 • Rédigez en Caml une fonction de signature

```
lien_mort : int -> table -> table
```

spécifiée comme suit : `lien_mort n t` construit, à partir de la table `t`, une nouvelle table mise à jour pour prendre en compte la mort du lien de numéro n.

Question 34 • Supposons que le lien 1 (reliant les routeurs A et B) soit coupé. Décrivez ce qui se passe ; on ne vous demande pas tous les détails. Constatez que le réseau converge vers un nouvel état stable satisfaisant les conditions établies aux question 26 et 27.

▶ Nous allons voir que le réseau peut diverger suite à des incidents. Supposons que les liens 1 et 5 soient coupés simultanément. Le routeur A constate la coupure, met à jour sa table de routage et l'expédie à C ; en revanche, C ne constate pas tout de suite la coupure, et expédie à A sa table de routage (laquelle mentionne, par exemple, que C connaît une route de longueur 2 menant à B).

Question 35 • Montrez que le sous-réseau constitué par A et C va diverger, même après la constatation par C de la coupure du lien 5. Plus précisément, vous montrerez que, dans les tables de A et C, vont apparaître des routes inexistantes, et de longueurs croissantes avec le temps.

▶ Le phénomène que nous venons de mettre en évidence est baptisé ≪comptage vers l'infini≫. Un premier remède consiste à décider que toute route de longueur supérieure à un seuil δ fixé est de longueur infinie. Dans l'Internet, $\delta = 15$.

Question 36 • Décrivez ce qui se passe, avec cette amélioration.

Question 37 • Discutez les avantages et inconvénients qu'il y aurait à augmenter ou diminuer le seuil δ.

▶ Un deuxième remède, connu sous le nom d'≪horizon partagé≫, s'appuie sur la remarque suivante : si un routeur x connaît une route menant à un routeur z, et dont la première étape passe par le routeur y (qui est donc l'un des voisins de x), il est inutile d'informer y de cette route.

Question 38 • Justifiez cette remarque et expliquez pourquoi elle empêche la divergence mise en évidence à la question 35.

▶ Pour terminer, nous examinons une amélioration du remède précédent, connue sous le nom d'≪horizon partagé avec empoisonnement des routes≫ : si un routeur x connaît une route menant à un routeur z, et dont la première étape passe par le routeur y (qui est donc l'un des voisins de x), alors x indique à y qu'il connaît une telle route, mais en lui donnant une *longueur infinie*.

Question 39 • Décrivez l'effet de la méthode ≪horizon partagé avec empoisonnement des routes≫, dans le scénario de coupure simultanée des liens 1 et 5.

FIN