

Réseaux pair à pair non structurés

Épreuve pratique d'algorithmique et de programmation

Concours commun des écoles normales supérieures

Durée de l'épreuve: 3 heures 30 minutes

Juillet 2009

ATTENTION !

N'oubliez en aucun cas de recopier votre u_0
à l'emplacement prévu sur votre fiche réponse

Important.

Sur votre table est indiqué un numéro u_0 qui servira d'entrée à vos programmes. Les réponses attendues sont généralement courtes et doivent être données sur la fiche réponse fournie à la fin du sujet. À la fin du sujet, vous trouverez en fait deux fiches réponses. La première est un exemple des réponses attendues pour un \tilde{u}_0 particulier (précisé sur cette même fiche et que nous notons avec un tilde pour éviter toute confusion!). Cette fiche est destinée à vous aider à vérifier le résultat de vos programmes en les testant avec \tilde{u}_0 au lieu de u_0 . Vous indiquerez vos réponses (correspondant à votre u_0) sur la seconde et vous la remettrez à l'examineur à la fin de l'épreuve.

En ce qui concerne la partie orale de l'examen, lorsque la description d'un algorithme est demandée, vous devez présenter son fonctionnement de façon schématique, courte et précise. Vous ne devez en aucun cas recopier le code de vos procédures!

Quand on demande la complexité en temps ou en mémoire d'un algorithme en fonction d'un paramètre n , on demande l'ordre de grandeur en fonction du paramètre, par exemple: $O(n^2)$, $O(n \log n)$,...

Il est recommandé de commencer par lancer vos programmes sur de petites valeurs des paramètres et de *tester vos programmes sur des petits exemples que vous aurez résolus préalablement à la main ou bien à l'aide de la fiche réponse type fournie en annexe*. Enfin, il est recommandé de lire l'intégralité du sujet avant de commencer afin d'effectuer les bons choix de structures de données dès le début.

Attention !

Certaines des questions de cette épreuve peuvent demander des calculs importants (de l'ordre de la dizaine de minutes). Il vous est donc fortement conseillé de lancer ces calculs une fois que vous êtes certain de la validité de vos fonctions et de passer aux questions suivantes pendant que l'ordinateur continue de calculer.

D'autre part, pour préparer votre réflexion sur certaines des questions à développer à l'oral, il pourra vous être utile d'utiliser les fonctions que vous aurez écrites pour expérimenter sur d'autres valeurs que celles précisées dans le sujet.

1 Introduction

Les systèmes informatiques actuels permettent à un grand nombre d'ordinateurs de communiquer via un réseau et de partager simplement des données.

L'architecture client/serveur désigne un mode de communication entre plusieurs ordinateurs d'un réseau qui distingue un ou plusieurs postes clients du serveur : chaque logiciel client peut envoyer des requêtes (demande d'une page Web, d'un mail, d'un fichier, d'un flux vidéo, ...) à un serveur (voir Figure 1(a)). Un serveur est donc initialement

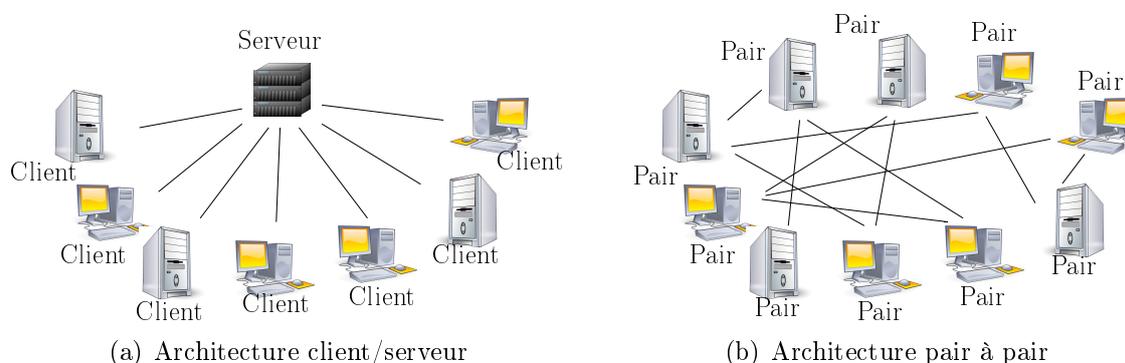


FIG. 1 – Architecture client/serveur et architecture pair à pair

passif, à l'écoute, et prêt à répondre aux requêtes envoyées par des clients. Dès qu'une requête lui parvient, il la traite et envoie une réponse. Un serveur est généralement capable de servir plusieurs clients simultanément.

Un client envoie des requêtes au serveur puis attend et reçoit les réponses du serveur.

Ce type d'architecture bien que très courant représente deux inconvénients majeurs :

- Lorsque le nombre de clients devient important, le serveur n'arrive plus à soutenir la charge des requêtes envoyées par les clients et son temps de réponse devient alors inacceptable.
- Si jamais le serveur tombe en panne, l'ensemble du système est bloqué.

L'architecture pair à pair a permis une décentralisation des systèmes, auparavant basés sur quelques serveurs, en permettant à tous les ordinateurs de jouer le rôle de client et serveur (voir Figure 1(b)).

En particulier, les systèmes de partage de fichiers permettent de rendre les objets d'autant plus disponibles qu'ils sont populaires, et donc répliqués sur un grand nombre de nœuds. Cela permet alors de diminuer la charge (en nombre de requêtes) imposée aux nœuds partageant les fichiers populaires, ce qui facilite l'augmentation du nombre de nœuds et donc de fichiers dans le réseau. C'est ce qu'on appelle le passage à l'échelle et concerne le premier inconvénient des architectures client-serveur précédemment évoqué.

Un autre avantage de ce type d'architecture par rapport aux architectures de type client-serveur est qu'un réseau pair à pair continue à fonctionner, même si plusieurs participants quittent le réseau. C'est ce qu'on appelle la tolérance aux pannes et permet de circonvier le second inconvénient des architectures client-serveur précédemment évoqué.

Ces systèmes ont connu un grand succès en facilitant la diffusion illégale de fichiers audio et vidéo mais le modèle pair à pair va bien plus loin que les applications de partage de fichiers. Il permet en effet de décentraliser des services et de mettre à disposition des ressources dans un réseau. De manière générale, les systèmes pair à pair permettent donc de faciliter le partage d'informations. Ils rendent aussi la censure ou les attaques légales ou pirates plus difficiles. Ces atouts font des systèmes pair à pair des outils de choix pour décentraliser des services qui doivent assurer une haute disponibilité tout en permettant de faibles coûts d'entretien. Toutefois, comme nous allons le voir, ces systèmes sont plus complexes à concevoir que les systèmes client-serveur.

Dans ce sujet, nous nous intéresserons plus particulièrement au problème de la diffusion d'information dans des réseaux pair à pair non structurés et à sa modélisation par le biais de graphes aléatoires.

Définition 1 (Graphe) *Un graphe est un couple $G = (S, A)$ avec S un ensemble de sommets et $A \subseteq S \times S$ un ensemble d'arêtes.*

En supposant les éléments de S arbitrairement numérotés s_0, \dots, s_{n-1} , un graphe peut se représenter par sa matrice d'adjacence notée a et définie par

$$a_{ij} = \begin{cases} 1 & \text{si } (s_i, s_j) \in A \\ 0 & \text{sinon} \end{cases}$$

On appelle degré d'un sommet le nombre de sommets avec qui il est en relation :

$$d(s_i) = \sum_{j=0}^{n-1} a_{ij}$$

Les sommets représentent les ordinateurs membres du réseau pair à pair (également appelés pairs) et les arêtes représentent les connexions entre les différents pairs. Lorsque deux pairs sont connectés, on dit qu'ils sont voisins. Ce graphe est non-orienté et non dégénéré, c'est-à-dire que A est une relation symétrique et non réflexive. La matrice s sera donc symétrique et à diagonale nulle. Dans la suite, les pairs sont numérotés de 0 à $n - 1$.

Le problème de la diffusion est le suivant :

Question 3 Calculer le degré du nœud d'indice u_0 des graphes suivants :

a) $G_0(100, 9)$ b) $G_2(1000, 55)$ c) $G_2(1000, 97)$.

Question 4 Calculer le degré moyen (c'est-à-dire $\frac{1}{n} \sum_{i=0}^{n-1} d(s_i)$) des graphes suivants :

a) $G_0(100, 9)$ b) $G_2(1000, 55)$ c) $G_2(1000, 97)$.

4 Calcul du diamètre

Maintenant que nous disposons d'une suite de graphes pseudo-aléatoires, nous allons pouvoir étudier quelques propriétés en rapport avec le problème de la diffusion.

Définition 4 (Chemin et distance) On appelle chemin de u à v une suite d'arêtes $(\gamma_0, \gamma_1), (\gamma_1, \gamma_2), \dots, (\gamma_{k-1}, \gamma_k)$ telle que $\gamma_0 = u, \gamma_k = v$ et les γ_i sont distincts entre eux. Un tel chemin est de longueur k .

On appellera distance la longueur du plus court chemin entre s et t . Par convention, la distance entre un sommet et lui-même est égale à 0. S'il n'existe aucun chemin entre deux sommets, on définira leur distance comme égale à $+\infty$.

Lors d'une diffusion initiée par un sommet source s , un sommet t obtiendra l'information si et seulement si il existe un chemin de s à t dans le réseau. S'il existe au moins un chemin, il obtiendra l'information au bout d'un temps proportionnel à la distance entre s et t .

La probabilité de succès d'une diffusion épidémique est donc lié à la probabilité que le graphe sous-jacent soit connexe (c'est-à-dire qu'il existe un chemin entre chaque paire de sommets). De même le temps de la diffusion est lié au diamètre δ du graphe, c'est-à-dire à la plus grande distance entre deux sommets.

Il est donc indispensable de savoir calculer la distance entre toutes les paires de sommets d'un graphe donné. Pour cela, on pourra utiliser la propriété suivante.

Propriété 5 Soit γ un chemin entre i et j de longueur minimale parmi l'ensemble des chemins entre i et j et dont les sommets intermédiaires sont dans $\{1, 2, 3, \dots, k\}$.

- soit γ n'emprunte pas le sommet k
- soit γ emprunte exactement une fois le sommet k (car les cycles sont de poids positifs ou nuls) et γ est donc la concaténation de deux chemins, entre i et k et k et j respectivement, dont les sommets intermédiaires sont dans $\{0, 1, 2, 3, \dots, k-1\}$.

Ainsi, si on note \mathcal{W}^k la matrice telle que \mathcal{W}_{ij}^k est la longueur minimale d'un chemin de i à j n'empruntant que des sommets intermédiaires dans $\{0, 1, 2, 3, \dots, k\}$, s'il en existe un, et ∞ sinon. Par extension, \mathcal{W}^{-1} est la matrice d'adjacence du graphe et l'observation ci-dessus se traduit par l'égalité suivante :

$$\mathcal{W}_{ij}^k = \min(\mathcal{W}_{ij}^{k-1}, \mathcal{W}_{ik}^{k-1} + \mathcal{W}_{kj}^{k-1}), \forall i, j, k \in \{0, 1, \dots, n-1\}.$$

En déduire un algorithme permettant de calculer la distance entre toutes les paires de sommets.

Question à développer pendant l'oral :

- Vous détaillerez la complexité de l'algorithme.

- Proposer un algorithme de complexité plus faible permettant uniquement de déterminer la connexité d'un graphe.
- Proposer un algorithme de complexité plus faible permettant uniquement de déterminer le diamètre du graphe.

Question 5 Indiquer si les graphes suivant sont connexes ou pas ainsi que leur diamètre le cas échéant :

$$\mathbf{a)} G_0(100, 10 \frac{u_{20}}{M}) \qquad \mathbf{b)} G_0(100, 10 \frac{u_{24}}{M}) \qquad \mathbf{c)} G_0(200, 20 \frac{u_{28}}{M}).$$

5 Effet de seuil connexité et diamètre moyen

On note

$$\mathcal{E}(n, d) = \{G_m(n, d) \text{ est connexe pour } m \in \{0, \dots, 49\}\}$$

On approxime la probabilité qu'un graphe soit connexe par

$$\mathcal{P}(n, d) = \frac{1}{50} \cdot \text{card}(\mathcal{E}(n, d)).$$

De même, on approxime l'espérance du diamètre d'un graphe sachant qu'il est connexe par

$$\mathcal{D}(n, d) = \frac{1}{\text{card}(\mathcal{E}(n, d))} \sum_{G \in \mathcal{E}(n, d)} \delta(G).$$

Question 6 Donner $\mathcal{P}(n, d)$ et $\mathcal{D}(n, d)$ pour $n = 100$ et $d = \ln(n) + c$, avec c variant de -2 à 5 .

Question 7 Donner $\mathcal{P}(n, d)$ et $\mathcal{D}(n, d)$ pour $n = 100$ et $d = c \cdot \ln(n)$, avec c variant de 1 à 3 .

Question à développer pendant l'oral :

- Qu'illustrent les expériences précédentes ?
- Que faudrait-il faire pour améliorer ces expériences ?
- En quoi ces propriétés sont-elles intéressantes dans le cadre des réseaux pair à pair ?

6 La composante géante

Définition 6 Étant donné un graphe G , on définit la relation d'équivalence \sim sur les sommets par $s \sim t$ si et seulement s'il existe un chemin entre s et t .

On appelle composante connexe une classe d'équivalence pour \sim .

Un sommet isolé est une composante connexe de cardinal 1 .

Dans cette section, on s'intéresse aux tailles des composantes connexes et en particulier à celle de la plus grande composante connexe.

Question 8 Calculer la taille de la plus grande composante connexe, le nombre de sommets isolés, ainsi que le nombre de composantes qui ne sont ni de taille maximale ni réduites à un seul sommet, pour les graphes suivants :

$$\mathbf{a)} G_0(100, 10 \frac{u_{20}}{M}) \qquad \mathbf{b)} G_0(100, 10 \frac{u_{24}}{M}) \qquad \mathbf{c)} G_0(200, 20 \frac{u_{28}}{M}).$$

On approxime l'espérance de la taille de la plus grande composante connexe par

$$\mathcal{C}(n, d) = \frac{1}{50} \sum_{m=0}^{49} \text{card}(\text{la plus grande composante connexe de } G_m(n, d)).$$

De même, on approxime l'espérance du nombre de sommets isolés par

$$\mathcal{I}(n, d) = \frac{1}{50} \sum_{m=0}^{49} \text{card}(\text{sommets isolés de } G_m(n, d)).$$

Enfin, on approxime l'espérance du nombre de composantes qui ne sont ni maximales, ni réduites à un seul sommet par

$$\mathcal{A}(n, d) = \frac{1}{50} \left(\sum_{m=0}^{49} \text{card}(\text{composantes connexes de } G_m(n, d)) \right) - \mathcal{I}(n, d) - 1.$$

Question 9 Donner $\mathcal{C}(n, d)$, $\mathcal{I}(n, d)$ et $\mathcal{A}(n, d)$ pour $n = 100$ et $d = \ln(n) + c$, avec c variant de -2 à 1 .

Question à développer pendant l'oral :

- Qu'illustrent les expériences suivantes ?
- Que faudrait-il faire pour améliorer ces expériences ?
- En quoi ces propriétés sont-elles intéressantes dans le cadre des réseaux pair à pair ?

7 Génération itérative de graphes aléatoire

Dans la suite, on dira qu'une condition est vraie avec probabilité p si en calculant le terme suivant de la suite v , c'est-à-dire le terme v_k si on a déjà utilisé les termes v_0, \dots, v_{k-1} auparavant, on a $v_k.p < N$.

Les graphes aléatoires d'Erdős et Rényi ont des propriétés très intéressantes mais assurer qu'un réseau pair à pair se structure de façon à ce qu'il puisse être modélisé par de tels graphes aléatoires est loin d'être évident. En effet, il faut pour cela que chaque pair rejoignant le réseau puisse se connecter à 1) en moyenne d voisins pris uniformément au hasard dans le réseau, 2) avec d dépendant de la taille du réseau. Ces 2 hypothèses sont difficiles à respecter puisqu'une des bases de cette architecture est justement que chaque pair ne connaît pas l'ensemble du réseau mais n'en voit qu'une petite partie.

On propose donc d'étudier l'algorithme d'agrégation suivant :

- Contact : un nouveau nœud s souhaitant rejoindre le réseau contacte *une* personne au hasard appelée *contact* parmi les pairs déjà existant. On s'assurera le nœud *contact* n'est pas un nœud isolé (le cas échéant, s contacte un autre nœud).
- Hello : lorsque le nœud *contact* est contacté par s , il fait suivre une demande d'inscription de la part de s avec probabilité $1/2$ à chacun des nœuds auxquels il est déjà connecté.

Il crée également c autres demandes d'inscriptions (c étant un paramètre de l'algorithme d'agrégation) et fait suivre chacune d'elle à un de ses voisins choisi aléatoirement.

– Inscription : lorsqu'un nœud n reçoit une demande d'inscription, une arête entre n et s est créée avec probabilité $p = \frac{1}{1+d(n)}$. Dans le cas où l'arête n'est pas créée, la demande d'inscription est renvoyée à l'un des voisins de n choisi aléatoirement.

Les demandes d'inscriptions ne disparaissent donc du réseau que lorsqu'une arête entre s et un autre nœud est créée.

On note $G'_m(n, c)$ le graphe obtenu en agrégeant $n - 20$ nœuds au graphe $G_m(20, 3c)$.

Question 10 *Calculer le degré moyen des graphes suivants :*

a) $G'_0(100, 1)$

b) $G'_0(100, 2)$

c) $G'_0(200, 2)$.

Question 11 *Évaluez l'espérance du degré moyen des graphes $G'(n, c)$ pour $n = 100$ et $c \in \{1, 2\}$ de la même façon que dans les sections précédentes (c'est-à-dire en effectuant une moyenne sur $\{G'_m(100, 2) | 0 \leq m \leq 49\}$).*

Question à développer pendant l'oral : Proposez une formule pour estimer l'espérance du degré moyen des graphes $G'(n, c)$. Vous pourrez tester votre hypothèse ou vous guider en utilisant les fonctions que vous avez précédemment utilisées.

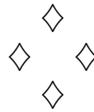
On définit $\mathcal{P}'(n, c)$ et $\mathcal{D}'(n, c)$ pour les graphes $G'(n, c)$ de la même façon qu'en section 5.

Question 12 *Donner $\mathcal{P}'(n, c)$ et $\mathcal{D}'(n, c)$ pour $n = 100$ et $c \in \{1, 2\}$.*

Question à développer pendant l'oral :

– Qu'illustre l'expérience précédente ?

– Quelle amélioration pourrait-on encore proposer à l'algorithme d'agrégation précédent ?



Fiche réponse type: Réseaux pair à pair non structurés

\widetilde{u}_0 : ... **5**

Question 1

a) **55175**

b) **35004**

c) **50225**

Question 2

a) **221 377 482**

b) **243 887 260**

c) **12 912 879**

Question 3

a) **6**

b) **54**

c) **105**

Question 4

a) **9.22**

b) **54.968**

c) **96.736**

Question 5

a) **non connexe**

b) **non connexe**

c) **connexe et diametre=6**

Question 6

n=100, c=-2 P(n,d)=0 D(n,d) (sans signification)

n=100 c=-1 P(n,d)=0.02 D(n,d)=7

n=100 c=0 P(n,d)=0.34 D(n,d)=6.47

n=100 c=1 P(n,d)=0.66 D(n,d)=5.67

n=100 c=2 P(n,d)=0.84 D(n,d)=4.97

n=100 c=3 P(n,d)=0.9 D(n,d)=4.4

n=100 c=4 P(n,d)=0.96 D(n,d)=4.17

n=100 c=5 P(n,d)=1 D(n,d)=4.04

Question 7

n=100 c=1 P(n,d)=0.34 D(n,d)=6.47

n=100 c=2 P(n,d)=1 D(n,d)=4.08

n=100 c=3 P(n,d)=1 D(n,d)=3.08

Notation des réponse Q8 et Q9.

Taille comp. connexe geante;

Nombre de sommets isoles;

Nombre des autres comp. connexes

Question 8

a) **91; 7; 1**

b) **95; 5; 0**

c) **200; 0; 0**

Question 9

n=100 c=-2 (90.56;7.38;0.86)

n=100 c=-1 (96.8;2.96;0.12)

n=100 c=0 (98.88;1.08;0.02)

n=100 c=1 (99.6;0.4;0)

Question 10

a)

b)

c)

Question 11

Question 12



Fiche réponse: Réseaux pair à pair non structurés

Nom, prénom, u₀:

Question 1

a)

b)

c)

Question 2

a)

b)

c)

Question 3

a)

b)

c)

Question 4

a)

b)

c)

Question 5

a)

b)

c)

Question 6

Question 7

Question 8

a)

b)

c)

Question 9

Question 10

a)

b)

c)

Question 11

Question 12

