

## Série 13

### 1 Couches et adresses

a) Pour chacune des adresses suivantes, indiquez à quelle couche du modèle TCP/IP elle est associée et quelle est sa portée (locale, globale ou processus).

Adresse	Exemple	Couche	Portée
f8:f2:1e:aa:cd:c0	adresse MAC	?	?
172.16.254.1	adresse IP	?	?
443	numéro de port	?	?

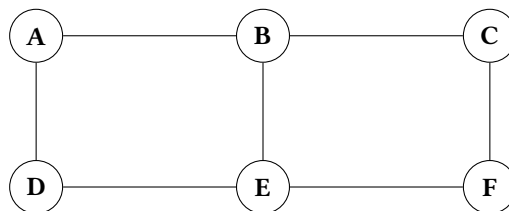
b) Bob, depuis son téléphone connecté en Wi-Fi, ouvre la page `https://epfl.ch`. À chacune des opérations ci-dessous, associez la couche TCP/IP (Application, Transport, Réseau, Liaison ou Physique) responsable :

1. Le navigateur construit une requête HTTP GET / à destination du serveur.
2. Un numéro de séquence et un accusé de réception sont ajoutés au paquet pour garantir une transmission fiable.
3. Le paquet est routé de proche en proche à travers plusieurs routeurs en suivant l'adresse IP de destination.
4. La carte Wi-Fi gère les collisions sur le canal partagé et réemet localement si besoin.
5. Les bits sont transmis sous forme d'ondes radio entre le téléphone et la borne Wi-Fi.

c) Lorsqu'un routeur *intermédiaire* relaie un paquet, jusqu'à quelle couche TCP/IP a-t-il besoin de remonter (*i.e.*, d'enlever les en-têtes)? Expliquez brièvement pourquoi il n'a pas besoin d'aller plus haut.

### 2 Routage et algorithme BFS

On considère le réseau à 6 nœuds suivant, dans lequel deux nœuds reliés par une arête sont séparés d'une distance de 1 :



a) En appliquant l'algorithme BFS (Breadth-First Search) depuis le nœud F, indiquez à côté de chaque nœud sa distance à F (autrement dit, le nombre minimum de sauts qui le séparent de F).

b) Donnez la table de routage complète du nœud A pour atteindre tous les autres nœuds du réseau. La table comporte trois colonnes : *destination*, *direction* (voisin direct par lequel passer) et *distance*. Indiquez clairement lorsque deux directions sont possibles.

c) Supposons maintenant que le lien B-E tombe en panne. Mettez à jour la table de routage de A : quelles entrées changent et pourquoi?

### 3 Protocole TCP : ACK, time-out et retransmission

Au cours, nous avons vu que dans le protocole TCP, un nœud A envoie un paquet à un nœud B, puis attend une notification (ACK) de B. Si cette notification n'arrive pas dans un délai  $T$  (*time-out*), A retransmet le paquet.

a) Citez les *trois* scénarios distincts qui peuvent amener A à retransmettre le paquet à l'expiration du temps  $T$  (donc trois raisons pour lesquelles l'ACK n'arrive pas à temps).

b) On suppose que la valeur de  $T$  a été mal choisie. Pour chacun des deux cas suivants, expliquez en quoi consiste le problème :

- $T$  est choisi *beaucoup trop grand*;

•  $T$  est choisi *beaucoup trop petit*.

c) Dans le cas où  $A$  retransmet alors que  $B$  avait en réalité bien reçu le premier paquet (mais que l'ACK s'est perdu ou est arrivé en retard),  $B$  va recevoir deux copies du même paquet. Comment  $B$  peut-il s'apercevoir qu'il s'agit d'un doublon et l'ignorer ?

## 4 Algorithme AIMD et autres

Au cours, nous avons vu que l'algorithme AIMD (*additive increase – multiplicative decrease*) permet de garantir un partage équitable du réseau lorsque deux utilisateurs  $A_1$  et  $A_2$  cherchent à communiquer de manière concurrente. Pour rappel, cet algorithme consiste, pour chaque utilisateur, à mettre à jour  $W$  (le nombre de paquets transmis par unité de temps) de la manière suivante :

$W \rightarrow W + a$  tant que tout se passe bien (avec  $a > 0$  un paramètre fixé);

$W \rightarrow b W$  dès qu'un problème survient (avec  $0 < b < 1$  un autre paramètre fixé).

En représentant l'évolution de  $W_1$  et  $W_2$  sur un même graphique (avec  $W_1$  en abscisse et  $W_2$  en ordonnée), nous avons vu en particulier que cet algorithme a pour effet, à terme, de rapprocher les valeurs de  $W_1$  et  $W_2$  et donc de permettre un partage équitable du réseau.

En réutilisant le même type de graphique avec  $W_1$  en abscisse et  $W_2$  en ordonnée, expliquez pourquoi *aucun* des trois algorithmes suivants ne permet d'atteindre ce but.

a) Algorithme AIAD (*additive increase, additive decrease*) :

$W \rightarrow W + a$  tant que tout se passe bien (avec  $a > 0$  un paramètre fixé);

$W \rightarrow W - \min(W, b)$  dès qu'un problème survient (avec  $b > 0$  un autre paramètre fixé).

b) Algorithme MIMD (*multiplicative increase, multiplicative decrease*) :

$W \rightarrow a W$  tant que tout se passe bien (avec  $a > 1$  un paramètre fixé);

$W \rightarrow b W$  dès qu'un problème survient (avec  $0 < b < 1$  un autre paramètre fixé).

c) Algorithme MIAD (*multiplicative increase, additive decrease*) :

$W \rightarrow a W$  tant que tout se passe bien (avec  $a > 1$  un paramètre fixé);

$W \rightarrow W - \min(W, b)$  dès qu'un problème survient (avec  $b > 0$  un autre paramètre fixé).

*Note* : Lors d'un retrait additif, on ne peut pas systématiquement soustraire une valeur fixe  $b > 0$  (au risque d'obtenir une valeur négative de  $W$ ); c'est pourquoi on soustrait  $\min(W, b)$  au lieu de  $b$ .

## 5 Questions d'examens passés

a)

Concernant le modèle en couches TCP/IP, quelles affirmations sont **vraies** ? (Plusieurs réponses peuvent être correctes.)

- Chaque couche ajoute (à l'envoi) son propre en-tête au paquet, et le retire (à la réception) pour transmettre la charge utile à la couche supérieure.
- C'est la couche Transport qui décide du chemin pris par les paquets à travers le réseau.
- L'adresse MAC est utilisée par la couche Liaison et n'a de sens qu'à l'échelle locale (au sein d'un même réseau).
- Un routeur intermédiaire dépaquette le message jusqu'à la couche Application pour pouvoir décider de la direction à emprunter.

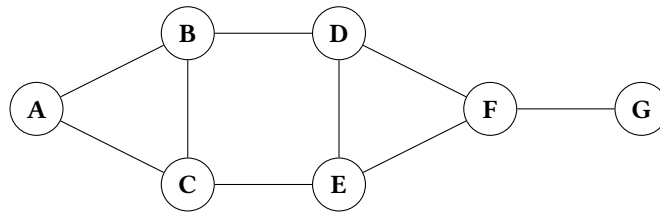
b)

Dans l'architecture TCP/IP, c'est la couche Transport qui décide du chemin pris par les paquets à travers le réseau.

- VRAI
- FAUX

c)

On considère le réseau suivant, dans lequel deux nœuds reliés par une arête sont séparés d'une distance 1 :



Parmi les affirmations suivantes, lesquelles sont **vraies**? (Plusieurs réponses peuvent être correctes.)

- La distance de A à G dans ce réseau vaut 4.
- Dans la table de routage de A, l'entrée pour la destination D admet deux directions possibles (B ou C).
- Si le nœud F tombe en panne, il n'existe plus aucune route pour atteindre G depuis le reste du réseau.
- Si on ajoute un lien direct entre C et F, la distance de A à G dans la nouvelle table de routage devient 3.

d)

On considère un réseau à 6 nœuds nommés A, B, C, D, E et F. On connaît (en partie) leurs tables de routage :

A		
dest	dir	dist
C	B	2
F	F	1

C		
dest	dir	dist
E	D	2
A	B	2

F		
dest	dir	dist
B	A	2
A	A	1

En vous basant uniquement sur le strict minimum de liens entre nœuds qui *doivent* exister selon les tables ci-dessus (*i.e.*, sans imaginer d'autres liens non-justifiés par ces tables) :

A. Dessinez le réseau le plus simple compatible avec ces informations.

B. Déduisez-en les valeurs de  $x$  et  $y$  dans l'entrée suivante de la table de routage de E :

E		
dest	dir	dist
A	$x$	$y$