

Exercice 1 :

Soit un fichier de 1Go¹. On veut le transmettre d'un site A à un site B distants de 6000 km. Les deux sites sont reliés par une connexion par fibre dont le débit est de 500 Mbits/seconde. On suppose que la vitesse de la lumière dans la fibre est de 2×10^8 m/s.

1. Si la connexion est sans erreur, combien de temps faut-il pour recevoir la totalité du fichier au site B ?

Dans la question précédente, nous avons ignoré le volume de données supplémentaires induit par les en-têtes (et checksums) des différentes couches. Supposons maintenant que le fichier est pris en charge par un protocole de couche application qui le découpe en groupes de 100 octets. Le groupe est ensuite encapsulé dans un segment UDP, puis un paquet IP, puis une trame Ethernet II (on ignore le préambule de synchronisation d'Ethernet). **On suppose qu'il n'y aucune fragmentation entre les couches.**

2. Quelle est la taille de chaque trame ?
3. En prenant en compte la taille de tous les en-têtes, quel volume de données est réellement envoyé ?
4. Combien de temps faut-il pour que le site B reçoive la totalité du fichier ? Même question si les groupes font 1000 octets au lieu de 100.

Supposons maintenant que la transmission n'est plus parfaite est que chaque bit a une probabilité d'erreur (BER) noté p . Gardons les groupes de 100 octets.

5. Quelle est la probabilité qu'une trame contienne au moins un bit erroné ? Application numérique : $p = 10^{-4}$.
6. Sachant qu'une trame contenant un bit erroné est systématiquement renvoyée, quel est le débit effectif moyen ?
7. Toujours en prenant en compte la taille des différents en-têtes, combien de temps faut-il en moyenne pour que B reçoive la totalité du fichier ?
8. Mêmes questions que 5. 6. et 7. En supposant que les groupes sont de 1000 octets et non pas 100.

Soit F la taille d'un fichier, p la probabilité d'erreur (BER) sur la liaison entre deux sites, D le débit de la liaison, e la taille des en-têtes ajoutés par toutes les couches (la somme). On suppose que le protocole de couche application découpe le fichier en groupes de g octets.

9. (**Question difficile**) En fonction des paramètres ci-dessus, donnez la valeur de g qui minimise le temps de transmission du fichier.

1 On suppose que 1Go = 10^9 octets.

Exercice 2 :

Soit la trace de trame Ethernet suivante capturée par Wireshark :

```

0000  01 00 5e 00 00 09 ca 02  06 d9 00 1c 08 00 45 c0  ..^.....E.
0010  00 34 00 00 00 00 02 11  17 2e c0 c0 00 02 e0 00  .4.....
0020  00 09 02 08 02 08 00 20  59 c0 01 02 00 00 00 00  .....Y.....
0030  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  .....
0040  00 10  ..

```

1. Donnez les adresses MAC source et destination.
2. Donnez le protocole de couche 3.
3. Donnez les adresses source et destination de couche 3.
4. Donnez le protocole de couche 4
5. Donnez les numéros de ports source et destination.
6. Quel est le protocole encapsulé dans celui de couche 4 ? Quelle est sa version ?
7. A quoi sert cette trame ?

Exercice 3 :

Soit la trace de trame Ethernet suivante capturée par Wireshark :

```

0000  ca 02 06 d9 00 1c ca 01  06 c6 00 1c 08 00 45 c0  .....E.
0010  00 40 00 1b 00 00 01 59  37 07 c0 c0 00 01 c0 c0  .@.....Y 7.....
0020  00 02 02 02 00 20 c0 c0  00 01 00 00 00 00 e0 5d  .....]
0030  00 00 00 00 00 00 00 00  00 00 05 dc 52 00 00 00  .....R...
0040  04 e2 ff f6 00 03 00 01  00 04 00 00 00 01  .....

```

1. Donnez les adresses MAC source et destination.
2. Donnez le protocole de couche 3.
3. Donnez les adresses source et destination de couche 3.
4. Quel est le protocole encapsulé dans celui de couche 3 ? Quelle est sa version ?
5. A quoi sert cette trame ?

Exercice 4 :

Le code de Hamming($m+k,m$) associe à un ensemble de m bits de données k bits de contrôle. Ce code permet de détecter et de corriger n'importe quelle erreur sur 1 bit. Le bit de données en position j est contrôlé par les bits dont les positions sont les coefficients de la décomposition binaire de j .

Nous avons vu le code Hamming(11,7) en TD, i.e., $m+k = 11$ et $m = 7$. Voyons maintenant le code Hamming(15,11) :

- [source] un mot de 11 bits ($s_3,s_5,s_6,s_7,s_9,s_{10},s_{11},s_{12},s_{13},s_{14},s_{15}$)
- [code] 4 bits de contrôle (c_1,c_2,c_4,c_8)
- [mot envoyé] 15 bits ($c_1,c_2,s_3,c_4,s_5,s_6,s_7,c_8,s_9,s_{10},s_{11},s_{12},s_{13},s_{14},s_{15}$)

1. Quels bits contrôlent s_{12} et s_{15} ?

Après codage et transmission, le mot reçu est le suivant : 110010111011010

2. Supposons qu'il n'y a eu qu'une seule erreur. Sur quel bit est-elle ?
3. Quel est le message source d'origine (corrigé) en binaire et en décimal ?
4. Pour un code de Hamming($m+k, m$), si k est fixe, quel est la valeur maximum possible de m ?