

# Sujet A : Allocation des canaux d'un réseau de bornes Wifi\*

## Notions abordées :

- Coloration de graphe (présence obligatoire TD SAé 3.02 lundi 12/12 8h-10h)

## Sujet :

Concevoir et développer un logiciel permettant, à partir d'un plan de positions des différentes bornes Wifi (coordonnées cartésiennes des positions) et de leur rayon de couverture, d'établir un canal d'affectation pour chaque borne.

## En entrée :

Les positions et rayons de couverture des bornes, à lire dans un fichier.

## Exemple :

1     200   50   20

2     -100   300   40

Signifie que la borne n° 1 est placée en ( $x=200$ ,  $y=50$ ) et possède un rayon de couverture de 20 m. La borne n° 2 est en ( $x=-100$ ,  $y=300$ ) et possède un rayon de couverture de 40 m.

## En sortie :

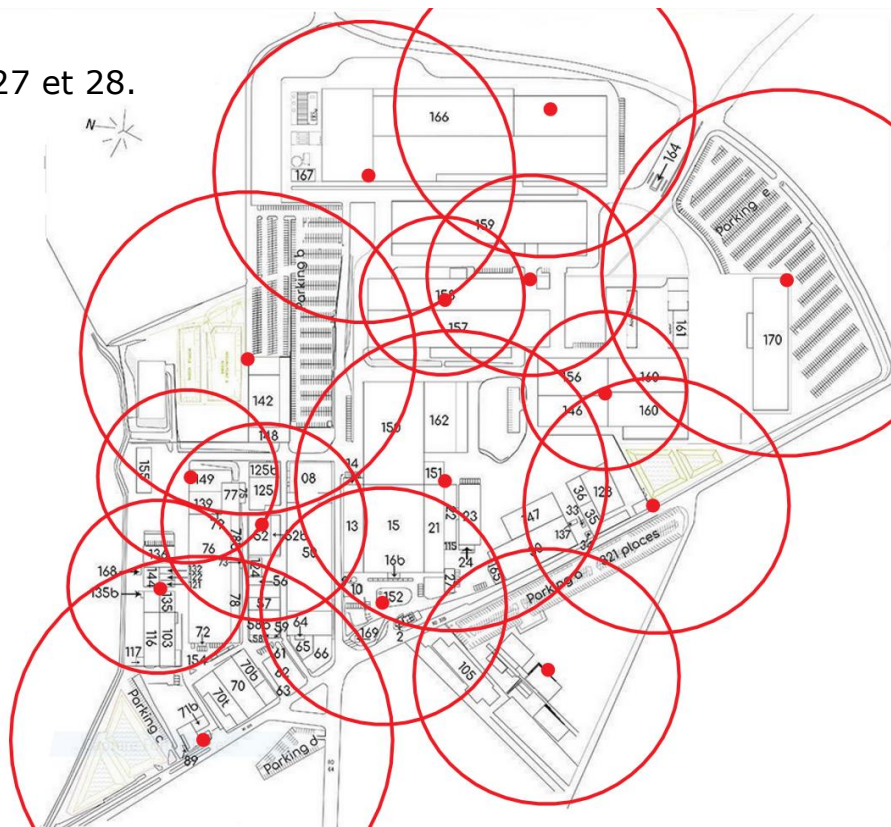
Un fichier de canaux d'affectation (à l'intention du technicien d'installation).

## Quelques précisions :

Le logiciel n'inclut pas de contrôle sur la pertinence de l'emplacement des bornes Wifi (ce contrôle est supposé fait en amont).

## Références :

- Cours pages 26, 27 et 28.



\* Technologie détaillée en BUT 3

# Sujet B : Répartition des charges dans un réseau redondant

## Notions abordées :

- Flot maximal dans un réseau de transport (présence obligatoire TD SAé 3.02 lundi 12/12 10h15-12h15)

## Sujet :

Concevoir et développer un logiciel permettant à partir d'une topologie redondante de switchs, de calculer le débit maximal du réseau, en provenance des utilisateurs (accédant aux switchs d'accès) en direction de la passerelle.

## En entrée :

Les caractéristiques de la topologie, à lire dans un fichier :

On lit dans un premier fichier les débits de chaque interface pour chaque commutateur : chaque ligne sera composée du numéro du commutateur suivi de couples indiquant le numéro de l'interface et son débit (en Mbit/s) et on terminera la ligne par un A (commutateur d'accès), C (commutateur de cœur de réseau) ou T (switch terminal, connecté avec le routeur passerelle)

Exemple :

Switch1 (eth1,100Mb) (eth2,1000Mb) (eth3,100Mb) Accès

Switch2 (eth1,1000Mb) (eth2,10000Mb) Cœur

signifie que le commutateur 1 est un commutateur d'accès. Il possède 3 interfaces connectées : la première est de débit 100Mbit/s, la 2<sup>ème</sup> 1000 Mbit/s et la 3<sup>ème</sup> 100Mbit/s. Le 2<sup>ème</sup> commutateur est dans le cœur de réseau et possède 2 interfaces connectées : la première est de débit 1000Mbit/s et la 2<sup>ème</sup> 10000 Mbit/s.

On lit dans un deuxième fichier les connexions filaires de la topologie : chaque ligne renseignera un lien. Elle sera composée de 2 couples : le premier indique le numéro du commutateur d'origine (ou 0 si c'est un lien d'accès qui connecte l'utilisateur à son switch d'accès) et le numéro de son interface connectée et le second le numéro du commutateur d'extrémité et son numéro d'interface.

Exemple :

(eth2,Switch1) (eth4,Switch3)

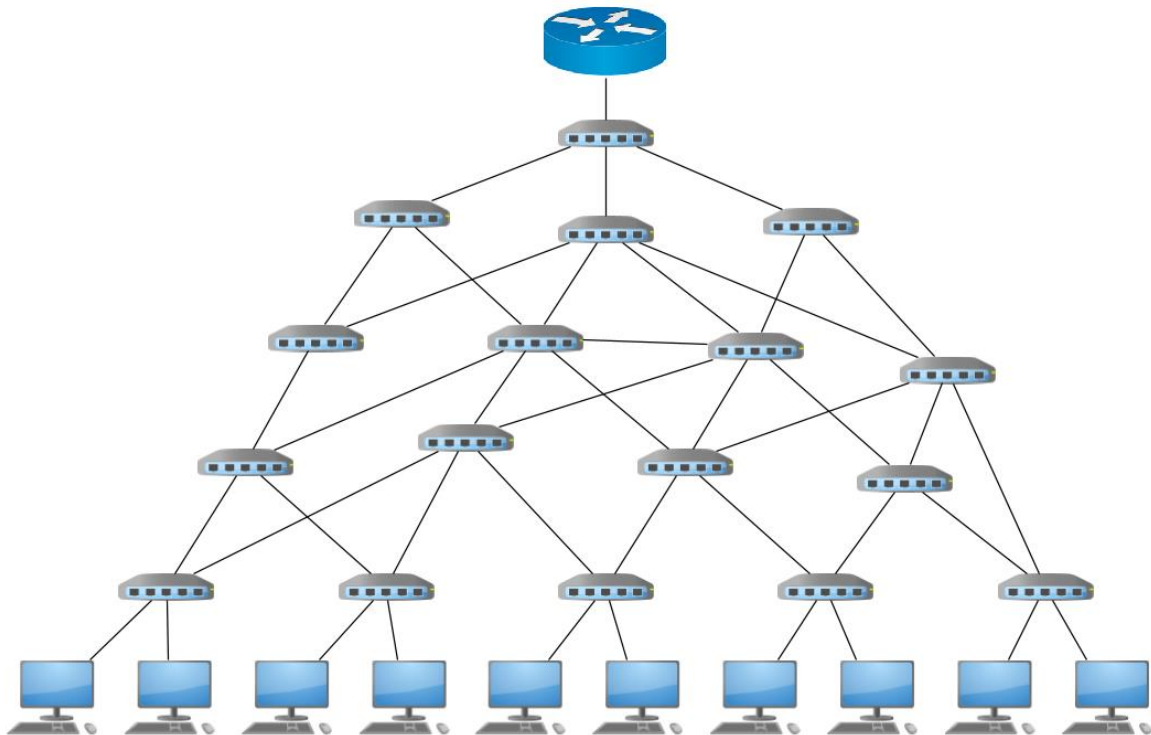
indique que l'interface eth2 du switch 1 est connectée à l'interface eth4 du switch 3.

## En sortie :

Le débit maximal du réseau (des switchs d'accès et le switch terminal) avec les encombrements de chaque lien. Une représentation graphique de la solution est la bienvenue (notamment les liens saturés).

## Références :

- Cours pages 29 à 32.
- Algorithmes 58 et 60 pages 142 et 144 de Eléments de théorie des graphes, éd. Lavoisier hermes, Alain Bretto (BU n°511.3 BRE).
- [https://perso.liris.cnrs.fr/samba-ndojh.ndiaye/fichiers/App\\_Graphes.pdf](https://perso.liris.cnrs.fr/samba-ndojh.ndiaye/fichiers/App_Graphes.pdf)



## **Sujet C : Positionnement du puits dans un réseau de capteurs sans fil**

### **Notions abordées :**

- Nœud médian dans un graphe

### **Sujet :**

Pour récupérer les informations recueillies par un réseau de capteurs sans fil, il faut désigner parmi eux un puits (capteur qui centralise les informations et les véhicule sur internet).

Pour communiquer les informations captées, chaque capteur dépense de l'énergie. Or pour chaque capteur, cette énergie est souvent très limitée (énergie solaire par exemple). D'où l'intérêt de positionner le capteur puits de façon à minimiser les dépenses d'énergie du réseau de capteurs.

Concevoir et développer un logiciel permettant, à partir d'une topologie de capteurs, de positionner le puits de façon à minimiser l'énergie dépensée en communication (on ne s'intéressera pas à l'énergie dépensée en acquisition d'information). Le logiciel indiquera également la quantité d'énergie moyenne dépensée par chaque capteur.

### **En entrée :**

Les coordonnées (x,y,d) de chaque capteur avec sa distance maximale d'émission, à lire dans un fichier.

Exemple :

1 (22,114,12)

signifie que le capteur 1 est positionné en (x=22, y=114) et possède une distance maximale d'émission de 12 mètres.

### **En sortie :**

La (ou les) position(s) idéale(s) du capteur-puits, avec l'énergie moyenne dépensée par chaque capteur (en supposant qu'une communication transitant par un capteur dépense une unité d'énergie, 1 UE).

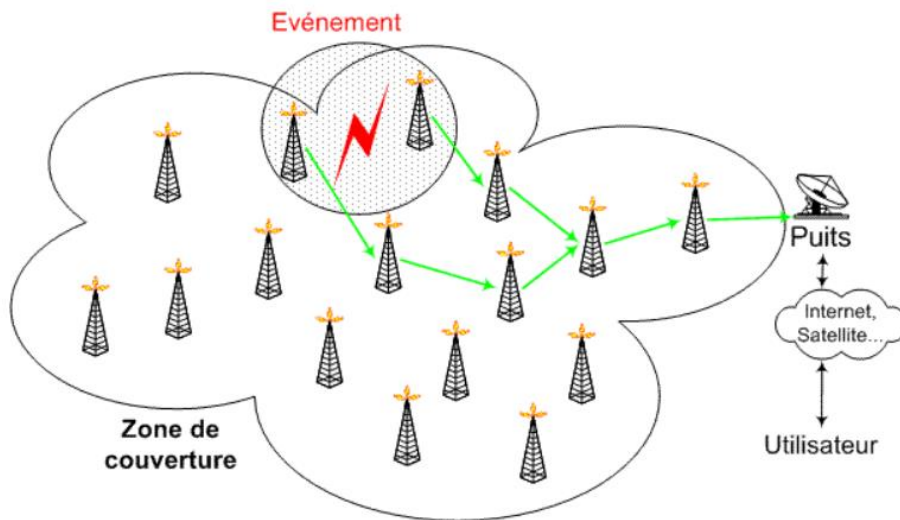
### **Quelques précisions :**

L'algorithme produit sert à l'étude de conception du réseau. On suppose qu'il est toujours possible de connecter chaque capteur à internet. Le capteur puits peut donc potentiellement être n'importe lequel des capteurs du réseau.

On supposera pour simplifier les choses que tous les capteurs sont du même modèle. Ils ont donc tous la même distance d'émission.

## Références :

- <http://igm.univ-mlv.fr/~dr/XPOSE2006/Bunel/Presentation.html>
- A la découverte des graphes et des algorithmes de graphes, C. Laforest, edp Sciences (BU n°511.3 LAF), chapitre 20 page 191.



## Sujet D : Multiplexage fréquentiel

### Notions abordées :

- Coloration de graphe (présence obligatoire TD SAé 3.02 lundi 12/12 8h-10h)

### Sujet :

Pour faire transiter simultanément plusieurs signaux dans la même fibre optique et les récupérer à la réception, il est essentiel que les gammes de fréquences des signaux soient disjointes. On désire multiplexer plusieurs signaux dont on connaît les gammes de fréquences dans un minimum de fibres.

Concevoir et développer un logiciel permettant, à partir des gammes de fréquences des signaux, lues dans un fichier, de donner le nombre minimum de fibres à utiliser et une répartition de multiplexage (quels signaux transitent dans quelle fibre).

### En entrée :

Les gammes de fréquences des signaux, à lire dans un fichier. Chaque ligne contiendra le numéro du signal suivi de sa fréquence minimale et de sa fréquence maximale (en kHz).

### Exemple :

Signal	fréq_min	fréq_max
1	100	500
2	300	700
3	750	1000

Signifie que le signal 1 a une gamme de fréquence de [100;500], le signal 2 a une gamme de fréquence de [300;700] et le signal 3 [750;1000].

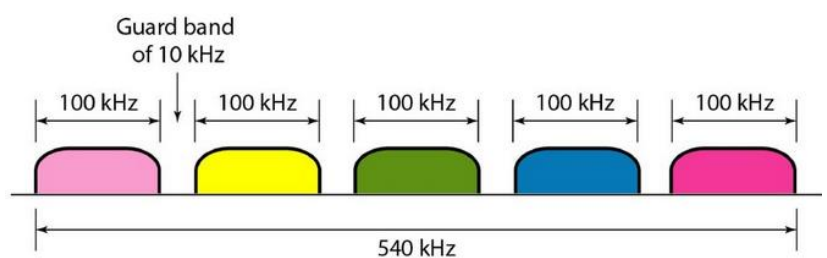
(Manifestement, ils ne pourront pas tous transiter dans la même fibre.)

### En sortie :

Le nombre de fibres à utiliser et les signaux cheminant dans chacune de ces fibres.

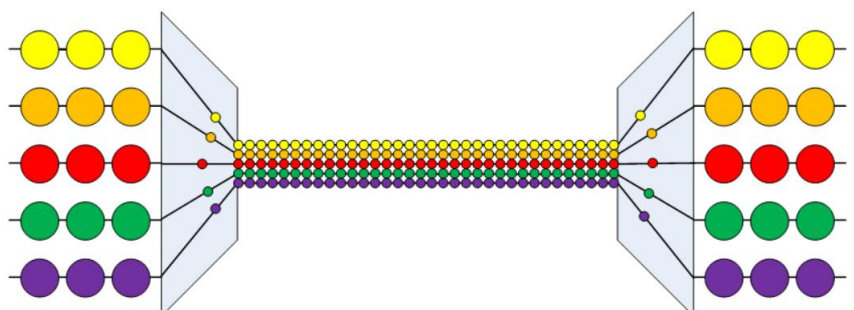
### Quelques précisions :

Pour plus de sûreté, on pourra laisser une marge (guard band) de 10 kHz entre deux gammes de fréquences.



### Références :

- Cours page 26 à 28





## Sujet E : Détermination du type d'AP Wifi, par zone

### Notions abordées :

- Plus court chemin dans un graphe

### Sujet :

Parmi les  $n$  AP d'un réseau Wifi, il y a  $k$  AP contrôleurs. Ces AP contrôleurs (de type C) communiquent entre eux les informations de contrôle du réseau. Chaque contrôleur communique aux AP de sa zone ces informations de contrôle. Les autres AP peuvent être de simple AP (de type S) destinées à la connexion des utilisateurs ou bien des AP Relais (de type R) permettant de relayer les informations de contrôle provenant des contrôleurs. Bien sûr, les caractéristiques (donc le coût) des trois types d'AP (C, R ou S) sont différents. C'est pourquoi il est déterminant de bien les distinguer.

On cherche à minimiser le trafic : pour cela, on affectera chaque AP (non contrôleur) à la zone du contrôleur le plus proche.

Développer un logiciel permettant, à partir des coordonnées  $(x,y)$  de chaque AP et de la liste des contrôleurs, de répartir les AP dans les zones dirigés par chaque contrôleur puis de statuer sur leur type (S ou R).

### En entrée :

Le fichier des noms et coordonnées de tous les AP. Chaque ligne correspond à un AP.

#### Exemple :

1 (12, 235) 15

2 (-150, 65) 25 indique que l'AP 1 a pour coordonnées  $(x=12, y=235)$  et un rayon de couverture de 15m et que l'AP 2 a pour coordonnées  $(x=-150, y=65)$  et un rayon de couverture de 25m.

Et le fichier contenant la liste des noms des contrôleurs.

#### Exemple :

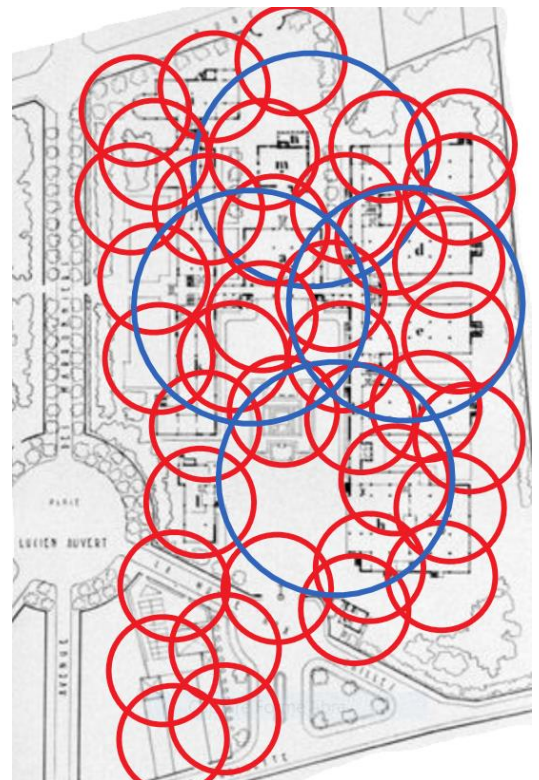
25 12 36 indique que les contrôleurs sont les AP numéros 25, 12 et 36.

### En sortie :

Un graphique de la topologie avec les AP colorés (suivant leur zone) et leurs types (joindre une légende au graphique).

### Quelques précisions :

Pour simplifier, on supposera que les rayons de couverture de tous ces AP sont les mêmes.



# Sujet F : Fiabilité d'un réseau

## Notions abordées :

- Graphe biconnexe
- Parcours en profondeur

## Sujet :

Un réseau est fiable s'il reste fonctionnel en cas de panne d'un équipement. En cas de panne d'un des deux routeurs, le routeur de secours prendra le relais.

Développer un logiciel permettant, à partir d'un réseau donné, de déterminer s'il est fiable et s'il est très fiable (panne de deux switchs simultanément).

## En entrée :

Le graphe non orienté, à lire dans un fichier, sous le format classique vu au TP1 de PROG4 (une ligne par équipement, y compris les deux routeurs).

## En sortie :

Edition des résultats : le réseau n'est pas fiable (indiquer pourquoi) ou il est fiable ou encore il est très fiable.

## Quelques précisions :

Attention, il faut que les deux routeurs soient toujours joignables.

## Références :

- <http://pauillac.inria.fr/~maranget/X/IF/02/index.html>
- <https://emre.me/algorithms/tarjans-algorithm/>
- <https://medium.com/@manujithaperera/finding-articulation-points-in-graphs-using-tarjans-algorithm-5ae253633ffe>

