

TD n° 1

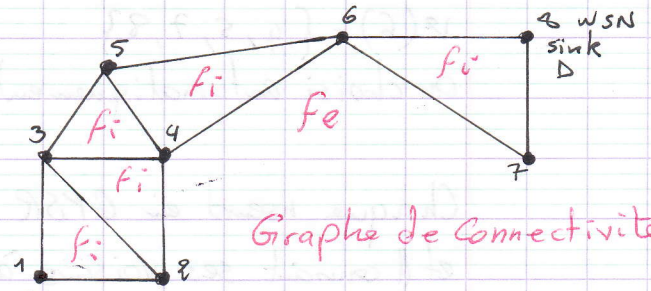
Exo 1 :

Perimeter Forwarding:
exige un graphe planaire

* Est-ce que le graphe est orienté ou non ? pourquoi ?

Si on considère une propagation réaliste des ondes radio, le graphe de connectivité est

considéré comme un graphe orienté, si on considère une propagation non réaliste (idéale) le graphe est orienté.

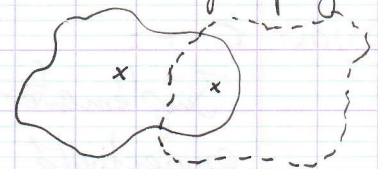


Graphe de Connectivité

WSN: wireless sensor network

fe: face externe

fi: face interne



Graphe planaire:

C'est un graphe où il n'y a pas deux arêtes qui s'intersectent.

* Quelle est la particularité du sink (RTS) ?

Nœud spécial qui n'a pas de contraintes d'énergie ni de ressources (CPU, ω , portée de transmission, bande passante). Il est relié à un centre de décision via un réseau haut débit.

Greedy Forwarding:

Le nœud x utilise un mode de communication multi sauts (multi-hops) veut transmettre au sink :

soit v l'un des voisins et $distE$ la distance Euclidienne $= \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$

$$v(8) = \{1, 3, 4\}$$

$$distE(1,8) > distE(3,8) > distE(4,8) \left\} \text{ donc on choisit le nœud 4}$$

$$distE(2,8) > distE(4,8)$$

$$V(4) = \{3, 5, 6, 8\}$$

$$\left. \begin{aligned} \text{dist}_E(3, 8) &> \text{dist}_E(5, 8) > \text{dist}_E(6, 8) \\ \text{dist}_E(4, 8) &> \text{dist}_E(6, 8) \end{aligned} \right\} \text{ on choisit nœud 6}$$

$$V(6) = \{4, 5, 7, 8\}$$

on choisit automatiquement le 8 puisque c'est lui le sink.

Chaque nœud en GFSR connaît la position du sink (coordonnées) et il connaît ses voisins à un saut donc il peut calculer la distance entre chaque voisin et le sink.

Si le mode Greedy échoue on passe au mode Perimeter

Devoir C++ :

Implémenter mode Greedy forwarding avec un graphe de connectivité où on peut trouver un vide de routage (mode greedy échoue).

Le graphe doit être généré aléatoirement

nombre de nœuds = 50

superficie = 30m / 30m

portée radio = 5m

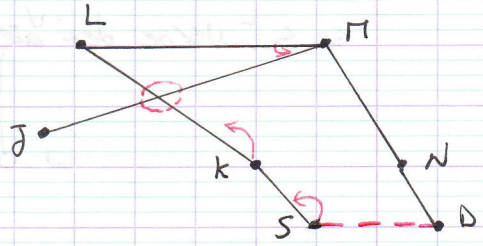
sink position : au centre

le programme doit afficher les voisins de chaque nœud avec leur position.

"échec, passage au mode périmètre" en cas d'échec.

Exo 2 =

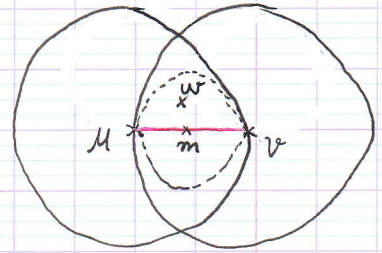
Il existe 2 algorithmes de planarisation, algorithme de Gabriel et RNG (Relative Neighbour Graph).



Algorithme de Gabriel =

fonctionne très bien ds les conditions idéales du canal de communication (Modèles de Radio de propagation non réalistes tel que Free space).

s'il existe au moins un nœud témoin w (witness) dans le cercle du rayon $\frac{uv}{2}$ alors le graphe n'est pas un graphe de Gabriel (planaire)



Pour tout $v \in \text{voisins}(u)$

Pour tout $w \in \text{voisins}(u) - \{v\}$

si $\text{dist}(m, w) > \text{dist}(m, v)$

(alors le graphe est planaire (graphe GG))

donc $uv \in GG(u)$

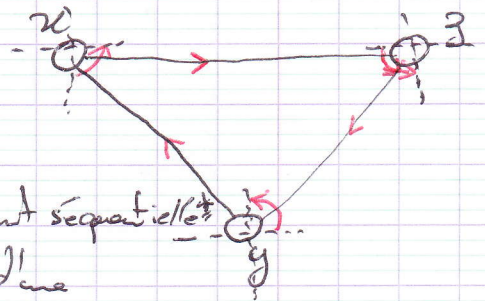
Règle de main droite :

Stipule que quand un msg arrivant au nœud z à partir de x ,

le prochain arc traversé est le suivant séquentiel dans le sens contraire des aiguilles d'une

montre à partir de l'arc xz

la face interne est traversée dans le sens des aiguilles d'une montre



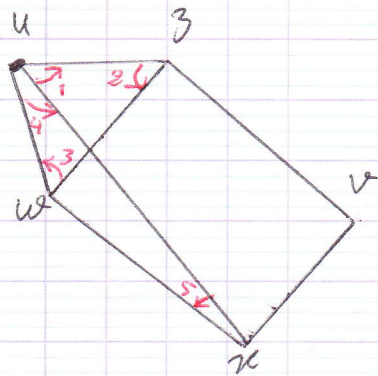
TD n° 3

Exo 1: Soit le graphe de connectivité suivant:
donner la règle de la main droite lorsque
 x envoie un paquet à u :

$x \rightarrow u \rightarrow z \rightarrow w \rightarrow u \rightarrow x \rightarrow w$

La règle de la main gauche:

$x \rightarrow u \rightarrow w \rightarrow z \rightarrow u \rightarrow x$



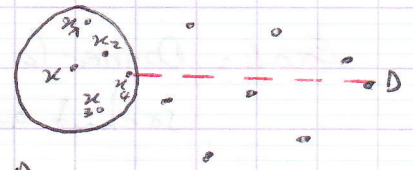
Exo 2: Un certain nbr de nœuds sont déployés
comme suit:

1/ donner les voisins de x : x_1, x_2, x_3, x_4

2/ donner le prochain saut à partir de x vers D en
mode Greedy: x_4

3/ Que constatez vous concernant le choix du next hop?

Le choix du prochain saut en utilisant la méthode Greedy basée uniquement sur la distance (prochain saut le plus proche de la dest par conséquent le plus éloigné du nœud source) a les désavantages suivants: de lien entre le nœud courant et le saut choisi peut être de faible signal (mauvaise qualité de lien \Rightarrow non fiabilité du lien).

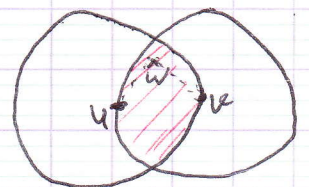
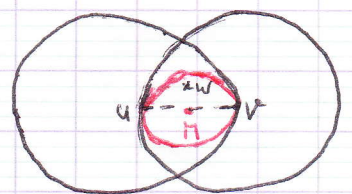


Exo 3: 1/ Rappeler la formule mathématique
qui permet de construire le GG:

$$MW > MV$$

2/ Trouver la formule mathématique
qui permet de construire le RNG:

$$\text{dist}(u, v) < \max(\text{dist}(u, w), \text{dist}(v, w))$$



3/ Donner l'algorithme:

Pour tout $v \in \text{voisins}(u)$ faire

pour tout $w \in \text{voisins}(u) - \{v\}$ faire

si $\text{dist}(u,v) > \max(\text{dist}(u,w), \text{dist}(v,w))$ alors

éliminer(uv)

break

fin

pour

pour

Exo 4: Donner la complexité des algorithmes de planarisation (GG, NRG) sachant que le nbr de voisins d'un nœud est appelé degrés