

Projet des graphes

BOUCHAMA Ayoub, ELGUERRAOUI Oussama

Contents

1	Introduction	3
2	Représentation des données de l'essaim sous forme de graphe	4
2.1	Coloration	4
2.2	Comparaison des données	6
2.2.1	Coefficient de clustering	6
2.2.2	Degré moyen	7
2.3	Composantes connexes d'ordre supérieur à 2	7
2.3.1	Composantes connexes du graphe Graph low 20km	7
2.3.2	Composantes connexes du graphe Graph low 40km	7
2.3.3	Composantes connexes du graphe Graph low 60km	7
2.3.4	Composantes connexes du graphe Graph average 20km	7
2.3.5	Composantes connexes du graphe Graph average 40km	8
2.3.6	Composantes connexes du graphe Graph average 60km	8
2.3.7	Composantes connexes du graphe Graph high 20km	8
2.3.8	Composantes connexes du graphe Graph high 40km	8
2.3.9	Composantes connexes du graphe Graph high 60km	8
3	Calcul du court chemin	9
4	Conclusion	9

List of Figures

1	Essaim de nanosatellites	3
2	Graphe Low Density, Range = 20km	4
3	Graphe Low Density, Range = 20km With Zoom	4
4	Graphe Low Density, Range = 40km	5
5	Graphe Low Density, Range = 40km With Zoom	5
6	Graphe Low Density, Range = 60km	5
7	Graphe Low Density, Range = 40km With Zoom	6
8	Chemin entre le noeud 97 et 99 dans le graphe pondéré	9
9	Chemin entre le noeud 97 et 99 dans le graphe non pondéré	9

1 Introduction

L'exploration spatiale continue de repousser les frontières de la technologie, ouvrant de nouvelles opportunités pour des missions ambitieuses et novatrices. Dans le cadre de ces avancées, un essaim de nanosatellites s'apprête à être déployé en orbite autour de la Lune dans le but de développer une application d'interférométrie. Cette technologie promet une meilleure compréhension de notre satellite naturel en capturant des données précises et en permettant des analyses détaillées de sa surface et de son environnement.

La communication au sein de cet essaim est essentielle pour assurer le succès de la mission. S'appuyant sur un routage opportuniste, les nanosatellites échangent des données lorsqu'ils se trouvent à portée les uns des autres, créant ainsi un réseau dynamique et auto-organisé. Pour optimiser cette communication, les protocoles existants se concentrent sur l'optimisation du taux de livraison des paquets et la minimisation du temps de latence, assurant ainsi une transmission efficace et rapide des données à travers le réseau.

Dans le cadre de ce projet, notre attention se porte sur trois moments spécifiques de l'essaim, correspondant à trois images capturées de celui-ci. Les données de mobilité de l'essaim de nanosatellites, fournies sous forme de fichiers CSV, permettent d'explorer les positions en trois dimensions de 100 satellites en fonction de la densité de l'essaim, à savoir une densité faible, moyenne et forte.

Dans ce contexte, notre objectif est d'analyser les caractéristiques de l'essaim dans ces trois configurations de densité. Nous nous intéressons notamment à la capacité des nanosatellites à moduler leur débit pour transmettre à différentes portées, ainsi qu'à l'obligation de transmettre les données de chaque satellite à tous les autres membres de l'essaim. Cette étude vise à fournir des insights précieux pour la conception et l'optimisation des systèmes de communication au sein d'essaims de nanosatellites, ouvrant ainsi la voie à de futures missions spatiales innovantes et performantes.



Figure 1: Essaim de nanosatellites

2 Représentation des données de l'essaim sous forme de graphe

2.1 Coloration

Notre approche pour visualiser les données de l'essaim à différentes densités consiste à les représenter dans un graphe non orienté et non pondéré. Nous nous assurons d'appliquer des règles de coloration afin d'éviter que deux sommets adjacents partagent la même couleur. Cette méthode permet une visualisation claire et précise des interactions entre les membres de l'essaim, soulignant les connexions et les relations entre les différents satellites.

Pour une portée de 20 km, nous constatons une connectivité limitée entre les différents sommets, ce qui suggère que les distances entre deux sommets sont supérieures à la portée spécifiée.

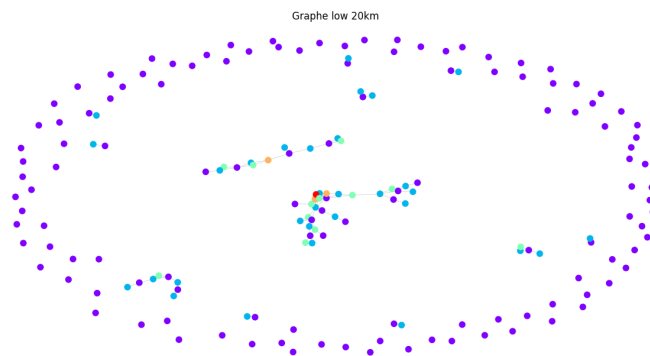


Figure 2: Graphe Low Density, Range = 20km

En zoomant sur l'ensemble compact des sommets au centre du graphe, on peut remarquer que la règle de coloration est bien présente.

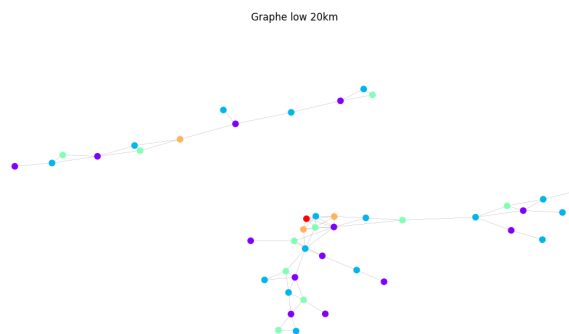


Figure 3: Graphe Low Density, Range = 20km With Zoom

Pour une portée de 40 km, nous constatons une connectivité plus riche par rapport à la précédente entre les différents sommets.

En zoomant sur l'ensemble compact des sommets au centre du graphe, on peut remarquer que la règle de coloration est bien présente.

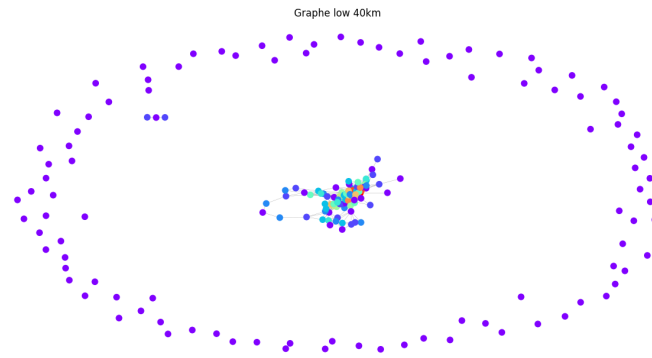


Figure 4: Graphe Low Density, Range = 40km

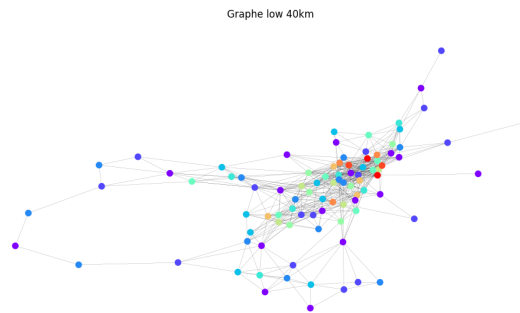


Figure 5: Graphe Low Density, Range = 40km With Zoom

Pour une portée de 60 km, nous constatons une connectivité très riche par rapport aux autres.

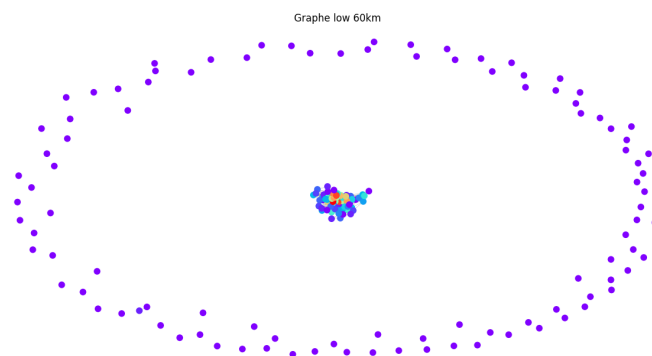


Figure 6: Graphe Low Density, Range = 60km

En zoomant sur l'ensemble compact des sommets au centre du graphe, on peut remarquer que la règle de coloration

est bien présente.

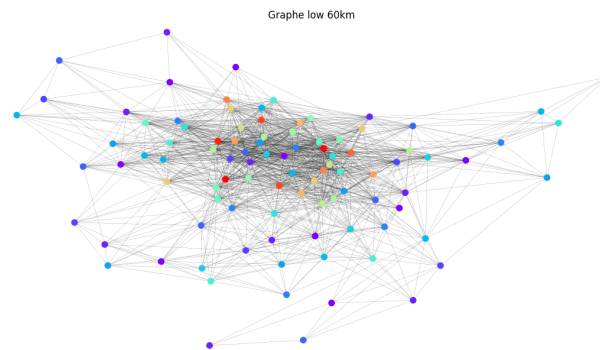


Figure 7: Graphe Low Density, Range = 40km With Zoom

Nous remarquons donc que la règle de coloration a bien été respectée et ceci également pour les autres type de densité.

2.2 Comparaison des données

Nous listons ici les données obtenues en calculant le degré de clustering global moyen de chaque graphe pour chaque densité et pour les trois portée et le degré moyen de chaque graphe.

Table 1: Comparaison des données

Graphe	Coefficient de clustering global		
	20 km	40 km	60 km
low	0.261	0.537	0.679
average	0.393	0.647	0.732
high	0.430	0.672	0.728

Table 2: Degrés moyens

Type	Degré moyen		
	20 km	40 km	60 km
low	1.040	5.887	14.859
average	1.870	8.548	17.910
high	2.011	9.434	18.700

2.2.1 Coefficient de clustering

La formule pour le coefficient de clustering local C_i d'un nœud i est la suivante :

$$C_i = \frac{2 \times \text{nombre de liens entre les voisins de } i}{\text{nombre total possible de liens entre les voisins de } i \times (\text{nombre total possible de liens entre les voisins de } i - 1)}$$

Un coefficient de clustering local proche de 1 indique que les voisins du nœud sont fortement connectés entre eux, ce qui suggère une structure de cluster ou de regroupement dans la région du graphe autour du nœud. Un coefficient de clustering local proche de 0 indique que les voisins du nœud ne sont pas fortement connectés entre eux.

On observe une tendance générale à une augmentation du coefficient de clustering global avec l'augmentation de la distance (de 20 km à 60 km) pour chaque type de graphe (low, average, high).

Les graphes à 60 km présentent les coefficients de clustering global les plus élevés pour chaque type, indiquant une plus grande propension à la formation de clusters locaux à mesure que la distance augmente.

Les graphes high ont tendance à avoir des coefficients de clustering globaux légèrement plus élevés que les graphes average, qui à leur tour sont légèrement plus élevés que les graphes low, suggérant une corrélation entre la densité du réseau et le coefficient de clustering global.

2.2.2 Degré moyen

Les degrés moyens (low, average, high) augmentent également avec la distance, ce qui est attendu car une distance plus grande signifie généralement une connectivité plus faible entre les nœuds et donc des degrés moyens plus élevés. Les degrés moyens les plus élevés sont observés pour le type de graphe high, ce qui pourrait indiquer une connectivité plus forte dans ces graphes par rapport aux graphes de type low et average.

Les différences entre les degrés moyens des types de graphes sont plus prononcées à 60 km, ce qui peut être dû à une plus grande dispersion des nœuds à mesure que la distance augmente.

2.3 Composantes connexes d'ordre supérieur à 2

Nous commençons par initialiser un ensemble vide pour stocker les composantes connexes trouvées et un ensemble vide pour suivre les nœuds déjà visités.

Ensuite, pour chaque nœud du graphe, nous vérifions s'il a déjà été visité. Sinon, nous commençons une nouvelle composante connexe. De plus, nous utilisons une fonction récursive de recherche en profondeur pour explorer tous les nœuds connectés au nœud actuel, en ajoutant chaque nœud visité à la composante connexe en cours de construction, et une fois que tous les nœuds connectés à la composante connexe actuelle ont été visités, nous ajoutons cette composante connexe à l'ensemble des composantes connexes trouvées.

On répète ce processus pour chaque nœud non visité dans le graphe, assurant que toutes les composantes connexes sont identifiées.

Une fois que toutes les composantes connexes ont été trouvées, nous renvoyons l'ensemble de ces composantes connexes.

2.3.1 Composantes connexes du graphe Graph low 20km

- Composante connexe : [1, 15, 74, 18, 52, 5, 84, 8, 46, 60, 48, 78, 68], Ordre : 13
- Composante connexe : [2, 50, 51, 7, 55, 73, 94, 67, 39, 42, 9, 30, 33, 23, 16, 38, 62, 69, 71, 6, 22, 34, 45, 72, 56, 90, 27, 97, 66, 80, 87], Ordre : 31

2.3.2 Composantes connexes du graphe Graph low 40km

- Composante connexe : [0, 21, 32, 57, 9, 6, 16, 5, 8, 14, 30, 22, 23, 27, 1, 15, 18, 20, 26, 35, 71, 33, 34, 42, 7, 19, 12, 63, 65, 66, 39, 50, 2, 11, 17, 24, 49, 10, 29, 28, 96, 43, 88, 31, 58, 55, 51, 60, 40, 36, 38, 41, 44, 48, 46, 52, 68, 67, 73, 78, 59, 98, 89, 64, 25, 13, 83, 94, 91, 82, 84, 74, 90, 45, 56, 62, 69, 61, 70, 75, 77, 79, 87, 80, 72, 93, 76, 97, 81, 95, 47], Ordre : 91

2.3.3 Composantes connexes du graphe Graph low 60km

- Composante connexe : [0, 12, 7, 2, 5, 1, 6, 8, 9, 14, 22, 16, 11, 17, 10, 24, 31, 58, 41, 18, 15, 20, 26, 23, 19, 4, 13, 25, 55, 30, 27, 33, 34, 35, 38, 36, 39, 40, 29, 28, 43, 37, 96, 74, 44, 46, 42, 45, 56, 21, 32, 57, 51, 48, 47, 59, 49, 85, 53, 99, 88, 50, 52, 60, 62, 61, 69, 65, 63, 66, 67, 68, 70, 71, 72, 77, 73, 64, 82, 83, 89, 78, 80, 75, 81, 87, 79, 95, 84, 90, 93, 76, 97, 94, 91, 98], Ordre : 96

2.3.4 Composantes connexes du graphe Graph average 20km

- Composante connexe : [0, 22, 90, 6, 16, 33, 18, 11, 1, 5, 15, 20, 30, 27, 42, 41, 44, 52, 7, 63, 9, 19, 2, 50, 71, 45, 66, 87, 51, 73, 60, 78, 80, 97, 74, 53, 40, 72, 57, 35, 84, 8, 17, 24, 48, 46, 64, 29, 98, 93, 32, 39, 12, 23, 62, 79, 38, 76, 65, 95, 70, 75, 77], Ordre : 63
- Composante connexe : [3, 26], Ordre : 2

- Composante connexe : [4, 13], Ordre : 2
- Composante connexe : [10, 49, 82, 94, 14, 55, 25, 67, 89, 91], Ordre : 10
- Composante connexe : [34, 43], Ordre : 2
- Composante connexe : [47, 59, 58, 88], Ordre : 4
- Composante connexe : [56, 69], Ordre : 2

2.3.5 Composantes connexes du graphe Graph average 40km

- Composante connexe : [0, 6, 2, 9, 7, 1, 5, 8, 11, 15, 18, 16, 20, 27, 3, 26, 34, 22, 12, 23, 32, 19, 39, 45, 30, 33, 35, 40, 28, 29, 17, 24, 44, 41, 42, 50, 51, 25, 4, 13, 67, 14, 10, 46, 48, 49, 55, 60, 52, 57, 53, 66, 63, 71, 65, 38, 21, 56, 62, 69, 76, 37, 61, 79, 95, 70, 68, 75, 77, 90, 87, 93, 43, 72, 74, 81, 84, 97, 73, 78, 80, 94, 82, 58, 47, 54, 59, 64, 85, 88, 89, 31, 91, 98, 99, 86, 96], Ordre : 97

2.3.6 Composantes connexes du graphe Graph average 60km

- Composante connexe : [0, 2, 1, 3, 5, 6, 9, 7, 8, 10, 4, 13, 14, 17, 11, 15, 16, 12, 18, 19, 20, 22, 21, 37, 23, 32, 27, 26, 28, 24, 25, 44, 29, 40, 30, 33, 34, 35, 41, 39, 38, 56, 43, 42, 45, 50, 51, 46, 47, 31, 49, 48, 52, 53, 57, 63, 55, 58, 54, 59, 64, 60, 66, 62, 65, 36, 70, 68, 67, 73, 71, 72, 74, 78, 80, 75, 77, 79, 69, 61, 76, 86, 95, 87, 90, 93, 84, 81, 82, 85, 88, 89, 91, 94, 98, 97, 99, 96, 83], Ordre : 99

2.3.7 Composantes connexes du graphe Graph high 20km

- Composante connexe : [0, 22, 90, 6, 16, 33, 18, 11, 1, 5, 15, 20, 30, 27, 42, 41, 44, 52, 7, 63, 9, 19, 2, 50, 71, 45, 66, 87, 51, 73, 60, 78, 80, 97, 74, 53, 40, 72, 57, 35, 84, 8, 17, 24, 48, 46, 64, 29, 98, 93, 32, 39, 12, 23, 62, 79, 38, 76, 65, 95, 70, 75, 77], Ordre : 63
- Composante connexe : [3, 26], Ordre : 2
- Composante connexe : [4, 13], Ordre : 2
- Composante connexe : [10, 49, 82, 94, 14, 55, 25, 67, 89, 91], Ordre : 10
- Composante connexe : [34, 43], Ordre : 2
- Composante connexe : [47, 59, 58, 88], Ordre : 4
- Composante connexe : [56, 69], Ordre : 2

2.3.8 Composantes connexes du graphe Graph high 40km

- Composante connexe : [0, 6, 2, 9, 7, 1, 5, 8, 11, 15, 18, 16, 20, 27, 3, 26, 34, 22, 12, 23, 32, 19, 39, 45, 30, 33, 35, 40, 28, 29, 17, 24, 44, 41, 42, 50, 51, 25, 4, 13, 67, 14, 10, 46, 48, 49, 55, 60, 52, 57, 53, 66, 63, 71, 65, 38, 21, 56, 62, 69, 76, 37, 61, 79, 95, 70, 68, 75, 77, 90, 87, 93, 43, 72, 74, 81, 84, 97, 73, 78, 80, 94, 82, 58, 47, 54, 59, 64, 85, 88, 89, 31, 91, 98, 99, 86, 96], Ordre : 97

2.3.9 Composantes connexes du graphe Graph high 60km

- Composante connexe : [0, 2, 1, 3, 5, 6, 9, 7, 8, 10, 4, 13, 14, 17, 11, 15, 16, 12, 18, 19, 20, 22, 21, 37, 23, 32, 27, 26, 28, 24, 25, 44, 29, 40, 30, 33, 34, 35, 41, 39, 38, 56, 43, 42, 45, 50, 51, 46, 47, 31, 49, 48, 52, 53, 57, 63, 55, 58, 54, 59, 64, 60, 66, 62, 65, 36, 70, 68, 67, 73, 71, 72, 74, 78, 80, 75, 77, 79, 69, 61, 76, 86, 95, 87, 90, 93, 84, 81, 82, 85, 88, 89, 91, 94, 98, 97, 99, 96, 83], Ordre : 99

3 Calcul du court chemin

- On a utiliser l'algorithme de dijkstra pour calculer le chemin le plus court entre un noeud i et un autre noeud j .
- On a implémenter une fonction `tableLongueur` qui permet de retourner une matrice dont le coefficient $a_{i,j}$ est la longueur du chemin entre le noeud i et le noeud j .
- On a également implémenter une fonction `tableChemin` qui retourne une liste de listes représentant le chemin correspondant entre le premier élément du tableau qui représente l'id du noeud source et le dernier élément du tableau qui représente l'id du noeud destination.
- Finalement, on a implémenté un algorithme qui retourne une matrice dont le coefficient $a_{i,j}$ représente le nombre de saut entre le noeud d'id i et celui d'id j .

Comparaison entre le chemin en graphe pondéré et non-pondéré

Nous avons remarqué que le chemin dans le graphe pondéré diffère complètement de celui du graphe non-pondéré vu que le poids de certaines arêtes peuvent élever la longueur du chemin, ce qui pousse à chercher une autre chemin.

Pour prendre un exemple concret de nos résultats, on a pris le chemin qui mène du noeud 97 vers le noeud 99.

```
Le chemin le plus court de 97 a 99 est : 97 59 27 33 48 25 29 11 24 79 61 99 -> Le nombre de saut est : 11
Longueur du chemin : 4150.925775474607
```

Figure 8: Chemin entre le noeud 97 et 99 dans le graphe pondéré

```
Le chemin le plus court de 97 a 99 est : 97 5 99 -> Le nombre de saut est : 2
Longueur du chemin : 2.0
```

Figure 9: Chemin entre le noeud 97 et 99 dans le graphe non pondéré

Nous remarquons que dans le graphe pondéré le nombre de saut est supérieur à celui du chemin dans le graphe non pondéré ce qui prouve que la pondération d'un graphe modifie complètement les chemins optimaux entre deux noeuds i et j .

Notamment, la longueur du chemin dans le graphe pondéré est plus de 2000 fois supérieur à celle du graphe non pondéré.

4 Conclusion

Ce projet a exploré les défis et les opportunités liés à la communication au sein d'essaims de nanosatellites en orbite lunaire, en se concentrant sur l'utilisation de données de mobilité pour analyser la connectivité et les performances des réseaux de communication. En utilisant des représentations graphiques et des techniques d'analyse de réseaux, nous avons examiné la connectivité, les clusters, les degrés moyens et d'autres mesures pour différentes densités et portées de communication.

Nous avons également implémenté et comparé plusieurs approches pour calculer le plus court chemin entre les nœuds du réseau, en utilisant à la fois des graphes pondérés et non pondérés. Cette comparaison a mis en évidence l'impact significatif de la pondération des graphes sur la détermination des chemins optimaux.

En conclusion, ce projet nous a fourni des insights précieux pour la conception et l'optimisation des systèmes de communication au sein d'essaims de nanosatellites, ouvrant ainsi la voie à de futures missions spatiales innovantes et performantes.