

Projet Ingénierie des réseaux

Clustering et routage dans un essaim de nano-satellites

Antoine Rey, Quentin Pointeau, Enzo Baïta, Evann Dreumont, Antonio-Alexandru Popescu-Cristea

29 mai 2025

Résumé—Dans ce projet, nous faisons l'état de l'art concernant le clustering et le routage dans les essaim de nano-satellites puis nous étudions l'efficacité d'algorithmes de clustering originaux dans le cadre d'un essaim de nano-satellites gravitant autour de la Lune.

Index Terms—IEEE, essaim de nano-satellites, cluster, cluterling, routage

I. ÉTAT DE L'ART

A. Introduction

Les essaims de nanosatellites sont des groupes constitués de petits satellites peu coûteux, ils sont de plus en plus utilisés pour des applications telles que l'observation de la Terre et la communication. Le clustering est une méthode qui permet de regrouper ces satellites pour une gestion efficace du réseau, tandis que le routage assure la transmission des données à travers l'essaim. Compte tenu de leurs contraintes en matière de ressources, ces deux domaines nécessitent des solutions innovantes.

B. Problèmes et solutions

Les nanosatellites sont confrontés à de nombreux problèmes tels que la mobilité, l'énergie et les défaillances fréquentes des liaisons. Les algorithmes de regroupement visent à former des groupes stables en sélectionnant les têtes de groupe en fonction de l'énergie et de la connectivité. Les protocoles de routage doivent permettre de trouver les meilleurs chemins dans des environnements dynamiques.

C. Clustering et routage dans un essaim de nanosatellites

Les essaims de nanosatellites, sont des groupes de satellites pesant moins de 10 kg. Étudions l'état de l'art en matière de clustering et de routage dans ces essaims, en regardant les développements récents et les solutions proposées.

1) *Caractéristiques des essaims de nanosatellites*: Les principales caractéristiques des essaims de nano-satellites sont les suivantes :

- *Orbite basse* : Les essaims de nanosatellites sont souvent déployés en orbite basse (low Earth orbit (LEO)).
- *Topologie dynamique* : Les satellites en orbite basse se déplacent à grande vitesse, ce qui entraîne des changements fréquents dans la topologie du réseau.
- *Contraintes de ressources* : L'énergie, la bande passante et les capacités de traitement limitées nécessitent des algorithmes légers et économes en énergie.

— *Évolutivité* : La gestion d'un grand nombre de satellites nécessite des algorithmes qui s'adaptent bien sans utiliser trop de ressources.

— *Fiabilité* : Les défaillances imprévisibles des liaisons et les dysfonctionnements des satellites exigent des solutions robustes pour garantir la stabilité du réseau.

2) *Algorithmes de clustering*: L'objectif de ces algorithmes est de regrouper les nanosatellites afin d'équilibrer la charge de traitement des données au sein de l'essaim.

— *Weight-based Dominating Set Clustering* : Cet algorithme sélectionne les têtes de groupes sur la base de poids associés à chaque nanosatellite en fonction de l'énergie et de la connectivité de celui-ci, ce qui garantit une formation efficace et stable des groupes [1]. Il est particulièrement adapté aux problèmes de densité et de taille des nano-satellites.

— *Agent-based Clustering Framework* : En 2021 des chercheurs ont introduit une approche dans laquelle les satellites sont modélisés comme des agents autonomes avec une structure hiérarchique, comprenant trois niveaux de contrôle pour la gestion des composants [2]. Ce cadre utilise un nouvel indicateur pour l'évaluation de l'efficacité, garantissant la stabilité et l'évolutivité, ce qui est important pour les essaims de nano-satellites dont les ressources sont critiques.

— *Virtual Agent Clustering* : Voici une autre approche qui propose un système de gestion de la mobilité utilisant des groupes d'agents virtuels (virtual agent clusters (VACs)) pour réduire les frais généraux et les délais de commutation de paquets dans les réseaux de satellites LEO [3]. Cette approche est pertinente pour les nanosatellites, où les transferts fréquents dus à une forte mobilité peuvent peser sur les ressources.

Pour les essaims de nano-satellites, ces algorithmes doivent être encore optimisés en termes d'efficacité énergétique, étant donné les limitations de puissance. Les objectifs sont de réduire la complexité de calcul et d'améliorer la fiabilité, ce qui est vital pour les nano-satellites.

Algorithmes	Caractéristiques	Avantages	Inconvénients
Weight-based Clustering	Trouve les têtes de groupe selon l'énergie et la connectivité	Efficace, stable, prend en compte l'énergie	Complexité dans les environnements dynamiques
Agent-based Clustering	Hiérarchique, agents autonomes	Évolutif, fiable, faible délai	Coût d'initialisation élevé

TABLE I

COMPARAISON DES ALGORITHMES DE CLUSTERING

3) *Algorithmes de routages*: L'objectif de ces algorithmes est d'acheminer l'information de manière fiable et doivent être adaptés en fonction des caractéristiques des nanosatellites. Ces algorithmes de routage doivent s'adapter à un fort temps de latence et optimiser la consommation d'énergie des nanosatellites [4].

- *Open Shortest Path First (OSPF)* : OSPF est un protocole de routage très utilisé pour des réseaux statiques. Cependant l'efficacité de ce protocole est limitée pour des réseaux dynamique comme les essaims de nanosatellites. Des travaux ce sont alors tournés vers une adaptation de ce protocole en milieu dynamique [4].
- *Delay-Tolerant Network (DTN)* : DTN est un protocole qui permet de s'adapter à la latence des réseaux ce qui est important dans le cas des essaims de nanosatellites. Des chercheurs ont montré qu'il était possible d'utiliser DTN pour les communications spatiales et une extension du DTN, appelée Deep Space Network (DSN) routing a également été appliquée aux nanosatellites [5].

Protocol	Latency	Connectivity	Energy Efficiency
OSPF	High	Medium	Low
DTN	Medium	High	Medium
Hybrid (OSPF + DTN)	Medium	High	High

TABLE II

RESILIENCE MATRIX OF ROUTING PROTOCOLS IN SATELLITE SWARMS [5]

Le tableau II nous montre une comparaison des deux principaux protocoles de routage pour les nanosatellites, on voit qu'on a une latence élevée, une connectivité moyenne et une efficacité énergétique faible pour OSPF à cause de la topologie dynamique du réseau. DTN améliore ces trois caractéristiques et est donc plus adapté. Cependant une approche hybride des deux protocoles permettrait d'améliorer d'avantage l'efficacité énergétique mais ce n'est encore qu'en phase de recherche.

D. Conclusion

Les essaims de nanosatellites présentent des défis majeurs en matière de gestion réseau, notamment à cause de leur mobilité, de la topologie dynamique et des ressources limitées. Le clustering permet d'optimiser l'organisation de l'essaim, tandis que le routage doit s'adapter aux interruptions de liens et à la latence. Les approches existantes, comme le clustering basé sur le poids ou les agents, et les protocoles comme DTN, offrent des solutions prometteuses, mais aucune n'est totalement satisfaisante. Le projet vise à explorer des méthodes de clustering en testant leur efficacité en terme de consommation d'énergie et de débit.

II. DONNÉES ÉTUDIÉES

A. Données brutes

Pour cette études nous avons utilisé comme données la trace d'un essaim de nanosatellites. Cela correspond à un jeu de données composé de 10 000 échantillons des coordonnées (x, y, z) de 100 nanosatellites.

B. Dynamique de l'essaim

Dans notre étude, les nanosatellites peuvent émettre à trois portées différentes : 20km, 40km et 60km. C'est sur ces valeurs que nous pouvons jouer pour améliorer l'efficacité de nos clusters. Donc, pour étudier la dynamique de l'essaim, nous avons configuré tous les nanosatellites à une portée de 60km et nous avons eu les résultats suivants :

Nombre moyen de cliques : 264.0665

Clique maximale de taille 32

Clique minimale de taille 3

Nombre moyen de composantes connexes : 2.2722

Nombre moyen de nano-satellites isolés : 0.9747

Taille moyenne de la composante connexe maximale : 98.4303

Nombre maximal de composantes connexes : 5

Avec une taille de clique maximale de 32, on remarque qu'il y a une grande partie des satellites qui sont proche les uns des autres et avec la taille moyenne de la composante connexe maximale, on remarque que même si tous les satellites émettent à la puissance maximale, il y en a au moins un en moyenne qui est isolé.

III. ALGORITHMES DE CLUSTERING

A. Introduction

Il nous a été demandé d'étudier des algorithmes de clustering dans le cadre des essaim de nanosatellites, il nous faut donc trouver des algorithmes qui permettent de trouver des clusters qui minimisent la consommation d'énergie et qui maximisent le débit.

B. Approche par composantes connexes

Nous avons commencé par une première approche simple qui se base sur les composantes connexes, on configure tous les satellites pour une portée de 20km et chaque composante connexe est un cluster. Si un satellite se retrouve seul on lui configure une portée de 60km pour pénaliser l'algorithme. Le pseudo-code de l'algorithme est le suivant :

- Configurer la portée de tous les nanosatellites à 60km
- Rajouter des liens de 20km quand c'est possible entre les différents nanosatellites
- Pour chaque nanosatellite qui a au moins un lien, configurer sa portée à 20km
- Calculer les composantes connexes pour avoir les clusters

C. Approche par k-moyennes

Nous avons remarqué que l'algorithme précédent nous donne sur tous les échantillons, un cluster composé de la plupart des nanosatellites au centre de l'essaim. Nous avons donc cherché un algorithme qui lui pourrait répartir les nanosatellites en clusters qui recouvrent la surface de l'essaim. Nous

avons donc pensé à l'algorithme des k-moyennes pour rassembler les nanosatellites par zone puis prendre un nanosatellite de chaque zone pour composer un cluster. Le pseudo-code de l'algorithme est donc le suivant :

- Configurer la portée de tous les nano-satellites à 60km
- Calculer les ensembles de voisins avec l'algorithme des k-voisins
- Tant que les ensembles ne sont pas vide
 - Prendre un satellite dans chaque ensemble pour former un cluster
- Pour chaque cluster, ajouter les liens quand c'est possible (20km, 40km 60km)
- Configurer la portée des satellites au maximum des liens qu'ils entretiennent avec leur cluster

IV. SIMULATION

A. Introduction

Comme énoncé plus haut nous voulons étudier l'efficacité de nos algorithmes de clustering, notamment leur consommation en énergie et leur débit. N'ayant pas eu de métrique précise à utiliser, nous avons décidé de modéliser la consommation d'énergie par le carré de la portée du satellite (par exemple si un satellite émet à une portée de 20km il consomme $20 \times 20 = 400$ unités d'énergie). Pour ce qui est du débit, nous l'avons modélisé par une fonction décroissante de la distance

- 20km donne un débit de 10Mb/s
- 40km donne un débit de 6Mb/s
- 60km donne un débit de 2Mb/s

Avec ces métriques on obtient les résultats suivants :

1) Algorithme des composantes connexes:

Puissance moyenne dans l'essai : 116982.4
 Débit moyen dans l'essai : 855.658
 Nombre moyen de nano-satellites isolés : 24.057
 Nombre moyen de clusters : 31.482
 Taille moyenne des clusters : 3.37
 Taille minimale des clusters : 1
 Taille maximale des clusters : 83

Avec cette technique, la plupart des satellites ont une portée configurée à 20km ce qui leur permet de ne pas consommer beaucoup d'énergie et d'avoir un débit élevé. Comme on peut le voir ici, on a une taille maximale de cluster qui est de 83 ce qui est très gros. On a aussi un nombre moyen de satellites isolés qui est de 24 ce qui veut dire que on a en moyenne au moins 24 clusters composés d'un seul satellite. Cette algorithme permet donc d'avoir une consommation basse en énergie et un bon débit mais il ne divise pas assez l'essai et ne permet donc pas de répartir la charge.

2) Algorithme basé sur les k-moyennes:

Puissance moyenne dans l'essai : 198803.6
 Débit moyen dans l'essai : 646.168

Nombre moyen de nano-satellites isolés : 27.379

Nombre moyen de clusters : 20.304
 Taille moyenne des clusters : 5.06
 Taille minimale des clusters : 1
 Taille maximale des clusters : 10

On se rend compte que la performance globale de cet algorithme est inférieure à celle de l'algorithme basé sur les composantes connexes. En effet, la construction des clusters par cet algorithme engendre des clusters où les nano-satellites au sein du cluster sont très éloigné. En outre, la plupart du temps les clusters ne forment pas une seule composante connexe, voire beaucoup de nano-satellites sont isolés et forment à eux-mêmes un cluster (en effet l'algorithme des k-moyennes ne renvoie pas forcément des ensembles de même taille et donc par la suite quand on construit nos clusters, quand il ne reste plus qu'un ensemble non vide, chaque satellite restant formera un cluster). Nous avons donc des nano-satellites qui sont très largement configurés pour émettre avec une portée de 60km donc avec un débit plus faible et une consommation d'énergie plus élevée. Cependant, il est à noter que le nombre moyens de clusters est plus faible et la taille des clusters plus grande ce qui implique ici une meilleure couverture dans l'espace de l'essai pour les clusters.

V. CONCLUSION

La recherche d'algorithmes de clustering pour des essais de nano-satellites est encore un sujet ouvert. Les algorithmes proposés dans la littérature sont encore expérimentaux et nous avons voulu au travers de ce projet vous présenter deux algorithmes originaux pour ce problème. Même si la comparaison avec les algorithmes proposés dans la littérature n'a pas pu se faire pour des raisons de temps et de complexité d'implémentation, nous avons pu néanmoins déterminer à notre échelle l'efficacité des algorithmes que nous vous avons proposé. En effet, l'algorithme basé sur les composantes connexes s'avère plus performant en termes de débit et d'efficacité énergétique. Cependant, l'algorithme basé sur les k-moyennes reste intéressant pour sa particularité de recouvrir plus également les nano-satellites dans l'espace délimité par l'essai. Les simulations ne sont certes pas parfaites mais nous espérons vous avoir donné une autre idée de ce que pourrait être les pistes à explorer quant au problème de clustering dans les essais de nano-satellites. Cela dit, les algorithmes proposés ne sont pas des algorithmes distribués. Une piste d'amélioration serait donc de rendre ces algorithmes distribués afin de de les rendre utilisable en pratique.

RÉFÉRENCES

- [1] J. LIU, X. ZHANG, R. ZHANG, T. HUANG et F. R. YU, "Reliable and Low-Overhead Clustering in LEO Small Satellite Networks," *IEEE Internet of Things Journal*, t. 9, n° 16, p. 14 844-14 856, 2022. DOI : 10.1109/IJOT.2021.3114423.

- [2] S. GENG, S. LIU, Z. FANG et S. GAO, “An agent-based clustering framework for reliable satellite networks,” *Reliability Engineering System Safety*, t. 212, p. 107 630, 2021, ISSN : 0951-8320. DOI : <https://doi.org/10.1016/j.ress.2021.107630>. adresse : <https://www.sciencedirect.com/science/article/pii/S095183202100171X>.
- [3] S. W. Y. ZHANG H. Li, “Reliable and Low-Overhead Clustering in LEO Small Satellite Networks,” *Journal of Satellite Communications and Networking*, t. 35, n° 16, p. 50-60, 2021.
- [4] X. ZHANG, K. SHI, S. ZHANG, D. LI et R. XIA, “Virtual Agent Clustering Based Mobility Management Over the Satellite Networks,” *IEEE Access*, t. 7, p. 89 544-89 555, 2019. DOI : 10.1109/ACCESS.2019.2926432.
- [5] L. BASSET, R. DHAOU, E. LOCHIN, P. BERNARD, D. PRADAS et B. TAURAN, “On Selecting a Routing Protocol for Nanosatellite Swarm Networks,” in *101st IEEE Vehicular Technology Conference*, Accepté pour publication dans IEEE VTC-Spring 2025 (Oslo Norvège), IEEE, Oslo, Norway, juin 2025. adresse : <https://hal.science/hal-04999739>.