



PROJET INGÉNIERIE DE RÉSEAUX

Résilience d'un essaim de nano-satellites

Deuxième année Département des Sciences du Numérique
Option : Télécommunications et Réseaux

Réalisé par : Yasmina
Abou-El-Abbas, Aicha ELIDRISSI,
Thomas Gilas

Encadré par :
Riadh Dhaou

Sommaire

1	Introduction	2
2	Problématique	2
3	Objectif du projet	3
4	État de l’art	3
4.1	Résilience et robustesse dans les réseaux de nanosatellites	3
4.2	Métriques de résilience et de robustesse	3
4.3	Approches de renforcement de la résilience	4
4.4	Algorithmes de division de graphe	4
4.5	Travaux connexes	4
5	Modélisation de l’essaim	4
5.1	Algorithmes de division de graphe	6
5.2	métriques pour évaluer la robustesse et la résilience du graphe	7
6	Méthodologie de Simulation	8
7	Visualisations	8
7.1	Visualisation des trajectoires orbitales	8
7.2	Visualisation des graphes du réseau à un instant donné	10
8	Analyse des résultats	12
8.1	Contexte expérimental	12
8.2	Robustesse :	12
8.3	Résilience :	13
8.4	Analyse statistique globale	14
8.5	Equité des groupes : RMSE	15
9	Conclusion	16

1 Introduction

Dans les architectures de communication spatiale modernes, les *essaims de nanosatellites* représentent une solution innovante et flexible pour assurer la couverture globale, la redondance et la résilience des réseaux. La transmission des données dans de tels réseaux repose souvent sur un *routage opportuniste*, où chaque satellite joue un rôle actif dans le relais et la diffusion des informations.

Cependant, la nature dynamique de ces essaims – marquée par la mobilité des satellites, l'évolution temporelle de la connectivité et la variabilité des distances – rend le réseau vulnérable à diverses *formes de pannes*, qu'elles soient prévisibles (décharge de batterie, fin de vie orbitale) ou non (défaillance matérielle, perturbations environnementales). Face à ces enjeux, il devient crucial de comprendre les mécanismes de **résilience**, c'est-à-dire la capacité du réseau à maintenir ses performances même en présence de perturbations.

S'inspirant du célèbre principe stratégique « **diviser pour mieux régner** », notre projet explore une approche fondée sur la division du réseau en sous-ensembles cohérents via des algorithmes de *division de graphe*. Cette démarche permet de localiser la diffusion de l'information, d'améliorer la tolérance aux pannes et de réduire la propagation des perturbations au sein du réseau global.

2 Problématique

Dans le cadre de notre projet, nous considérons **un essaim de 100 nanosatellites** où chaque satellite doit transmettre ses données à tous les autres membres. Le réseau est construit à partir de graphes temporels, où les liens entre satellites sont définis selon leur proximité spatiale (avec différentes portées de communication simulées : 20 km, 40 km et 60 km).

Or, les premières simulations (travail de l'année précédente) ont mis en évidence une **vulnérabilité importante du réseau** : la suppression de certains nœuds centraux peut entraîner une chute drastique de la connectivité.

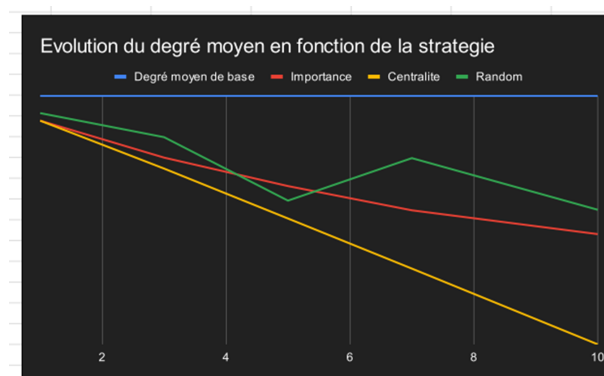


FIGURE 1 – Résultats du travail du groupe de l'année précédente

La problématique à laquelle nous répondons est donc la suivante :

Comment améliorer la résilience et la robustesse des communications dans un réseau dynamique de nanosatellites, soumis à des pannes aléatoires ou ciblées ?

3 Objectif du projet

L'objectif principal de ce projet est de :

- **Modéliser le réseau** de l'essaim sous forme de graphe dynamique pondéré ;
- **Analyser sa résilience** à travers un ensemble de métriques issues de la littérature scientifique ;
- **Proposer une méthode de renforcement de la résilience** par la division du graphe en groupes de satellites ;
- **Implémenter et comparer plusieurs algorithmes de partitionnement** (Random, Forest Fire, Random Walks) ;
- **Simuler différents scénarios de panne** (aléatoire ou ciblée) et mesurer les impacts ;
- **Évaluer les bénéfices et les limites** de chaque stratégie à travers une analyse des résultats.

4 État de l'art

4.1 Résilience et robustesse dans les réseaux de nanosatellites

Les réseaux d'essaims de nanosatellites constituent une architecture distribuée prometteuse pour des missions spatiales nécessitant couverture étendue, redondance et coût réduit. Toutefois, leur nature dynamique, les contraintes énergétiques, et la probabilité accrue de pannes posent des défis critiques en termes de fiabilité.

La **résilience** d'un réseau désigne sa capacité à maintenir ses fonctions même en présence de défaillances (nœuds ou liens). Elle est liée à la **robustesse**, qui mesure la résistance structurelle du réseau face à des perturbations. Ces notions sont fondamentales pour garantir la connectivité et la qualité de service dans un réseau spatial mobile.

4.2 Métriques de résilience et de robustesse

De nombreuses métriques ont été introduites dans la littérature pour quantifier ces deux propriétés, notamment :

- la robustesse de flux,
- la redondance des chemins,
- la disparité des chemins,
- la criticité des nœuds,
- le coût de routage,
- l'efficacité globale du réseau.

Ces métriques seront détaillées dans une section ultérieure. Elles constituent la base de l'évaluation des performances du réseau avant et après division, ainsi qu'en cas de panne simulée.

4.3 Approches de renforcement de la résilience

Différentes stratégies existent pour renforcer la résilience :

- Ajout de redondance (liens ou nœuds), souvent coûteux pour des nanosatellites.
- Réorganisation topologique dynamique selon les conditions du réseau.
- **Division du graphe** (ou partitionnement), permettant de structurer le réseau en sous-ensembles cohérents et robustes.

Cette dernière approche est celle retenue dans ce projet.

4.4 Algorithmes de division de graphe

Nous étudions trois algorithmes issus de la littérature :

- **RND (Random Node Division)** : assignation aléatoire des nœuds aux groupes.
- **FFD (Forest Fire Division)** : propagation probabiliste à partir de nœuds sources.
- **MIRW (Multiple Independent Random Walks)** : exploration par marches aléatoires indépendantes.

Ces algorithmes sont évalués à travers des simulations et comparés selon les métriques susmentionnées.

4.5 Travaux connexes

Plusieurs travaux récents ont abordé la résilience dans les systèmes satellitaires :

- **Akopyan (2024)** a proposé un cadre de simulation complet basé sur des métriques avancées.
- **Hanafi (2022)** a travaillé sur la fiabilité du nanosatellite universitaire MASAT-1.
- **Meziane-Tani (2018)** a exploré la faisabilité de constellations Cubesat pour la collecte de données.

Ces études soulignent l'intérêt croissant pour la structuration intelligente des essaims spatiaux, notamment via des techniques de partitionnement pour accroître leur résilience.

5 Modélisation de l'essaim

L'essaim de nanosatellites est modélisé par un **graphe temporel** $G_t = (V_t, E_t)$, où :

- V_t désigne l'ensemble des sommets (les nanosatellites) à l'instant t , de taille $n(t)$;
- E_t représente l'ensemble des arêtes, de taille $m(t)$, reliant les satellites.

Chaque arête est *pondérée* par un coût de communication dépendant de la distance euclidienne entre deux satellites. Pour simplifier le modèle, nous supposons que les liens sont **symétriques et duplex**, ce qui fait de G_t un *graphe non orienté pondéré*.

Nous considérons que chaque satellite souhaite communiquer avec tous les autres. Le *nombre total de communications possibles* (flow number), noté $NG(t)$, est donné par :

$$NG(t) = \frac{n(t) \cdot (n(t) - 1)}{2}$$

Ce nombre joue un rôle clé dans l'évaluation de la **résilience** et de la **robustesse** du réseau.

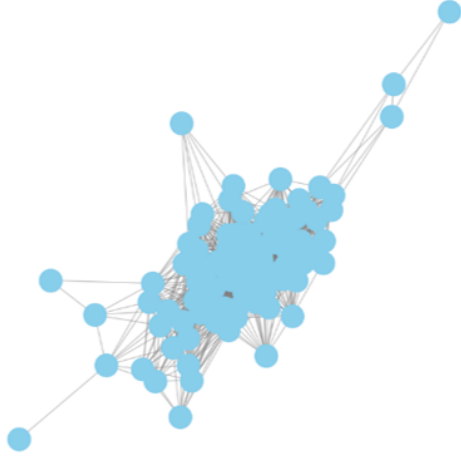


FIGURE 2 – le graphe temporel généré à $t=10$

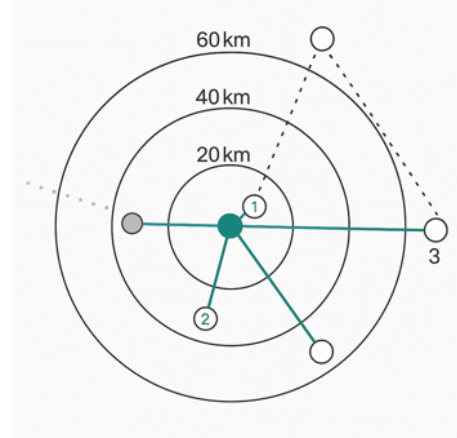


FIGURE 3 – poids selon portée

Effet de la division du graphe

Après application d'un algorithme de *division de graphe*, on obtient une nouvelle structure notée $G_t^* = (V_t^*, E_t)$, où :

- les sommets V_t sont répartis dans x groupes disjoints $\{V_1, V_2, \dots, V_x\}$;
- les arêtes E_t restent inchangées.

Dans ce cas, les transmissions sont structurées en deux catégories :

- **Intra-groupe** : chaque nœud communique d'abord avec les nœuds de son propre groupe. Le nombre total de ces communications est :

$$\sum_{i=1}^x \frac{n_i \cdot (n_i - 1)}{2}$$

- **Inter-groupe** : les communications entre groupes, approximées par :

$$\sum_{1 \leq i < j \leq x} n_i \cdot n_j$$

Ce nouveau modèle permet une communication localisée, améliore la **tolérance aux pannes** et réduit le coût global de transmission, tout en maintenant une certaine connectivité entre groupes.

Commentaire : Dans une configuration monolithique, où chaque satellite communique avec l'ensemble du réseau, le flow number quadratique $NG(t) = \frac{n(t) \cdot (n(t)-1)}{2}$ engendre une congestion inévitable et une vulnérabilité accrue aux défaillances en chaîne. La division du graphe en clusters hiérarchiques répond à cette problématique en structurant les communications selon deux niveaux : intra-groupe (optimisation locale) et inter-groupes (coordination globale). Cette approche réduit drastiquement le nombre de paires communicantes initiales tout en isolant les pannes au sein de groupes autonomes, limitant ainsi les perturbations à l'échelle du réseau. Théoriquement, elle s'appuie sur des principes de la théorie des graphes tels que la connectivité algébrique et le diamètre effectif, garantissant une redondance des chemins et une tolérance aux défaillances.

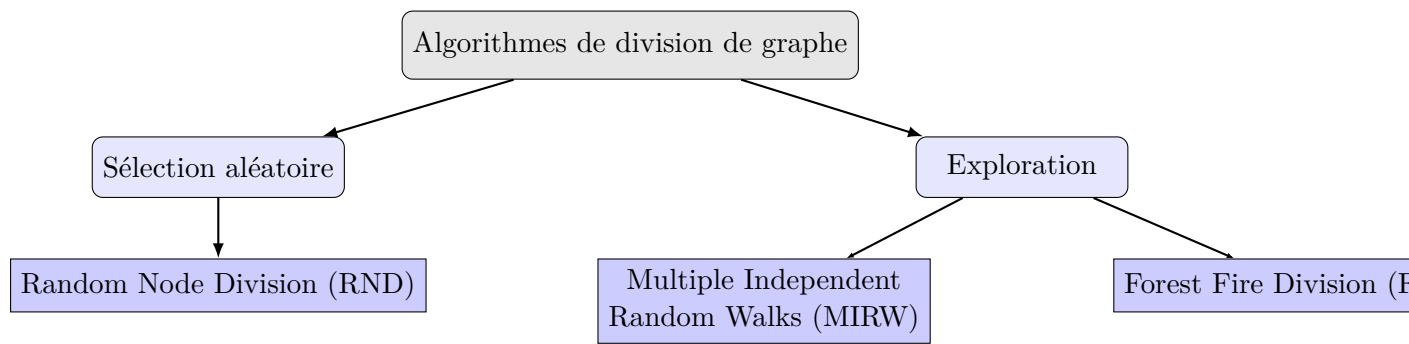
5.1 Algorithmes de division de graphe

Objectif de la division du graphe

La **division de graphe** (ou *partitionnement*) consiste à répartir les nœuds du réseau en groupes disjoints, appelés clusters ou sous-essaims. L'objectif est de réduire les coûts de communication en favorisant les échanges locaux, limiter l'impact des pannes à une portion restreinte du réseau, et améliorer la résilience locale, même si la robustesse globale peut légèrement diminuer.

Dans notre cas, le réseau de 100 nanosatellites est divisé en 5 à 10 groupes à l'aide de trois algorithmes que nous avons implémentés et comparés.

Algorithmes de Partitioning



On distingue principalement deux grandes familles d'algorithmes :

— Sélection aléatoire

Cette famille regroupe les méthodes qui attribuent les nœuds à des groupes de façon purement aléatoire, sans prendre en compte la structure du graphe.

Random Node Division (RND) : RND consiste à assigner chaque nœud à une partition choisie aléatoirement. Cette approche est simple et rapide à mettre en œuvre, mais elle peut conduire à des groupes déséquilibrés et à une faible cohésion interne, car elle ignore la connectivité réelle du réseau.

— Exploration

Ces méthodes exploitent la topologie du graphe pour constituer des groupes plus cohérents.

Multiple Independent Random Walks (MIRW) : MIRW repose sur plusieurs marches aléatoires indépendantes, chacune partant d'un sommet différent. Chaque

marche explore le graphe et assigne les sommets rencontrés à un groupe. Ce procédé garantit que tous les nœuds sont couverts, même dans les zones faiblement connectées, grâce à des « sauts » aléatoires en cas de blocage. MIRW est particulièrement adapté à la détection de communautés ou à la caractérisation de zones fortement connectées.

Forest Fire Division (FFD) : FFD s'inspire de la propagation d'un feu de forêt. À partir de nœuds sources, la propagation se fait de proche en proche, chaque voisin étant « enflammé » avec une certaine probabilité. Ce mécanisme produit des groupes compacts et connectés, et permet d'analyser la résilience locale du réseau. FFD se distingue de MIRW par son mode de croissance : alors que MIRW privilégie la diversité des parcours indépendants, FFD s'appuie sur une contagion contrôlée à partir de points initiaux.

L'implantation des algorithmes est donnée dans le fichier `swarm_sim.py`.

5.2 métriques pour évaluer la robustesse et la résilience du graphe

Robustesse	Résilience
<p>flow robustness / robustesse de flux : cette métrique mesure la proportion de paires de sommets connectées par un chemin dans le graphe, qu'il soit divisé ou non</p> $F_t(G) = \frac{f_t(G)}{N_G(t)}$	<p>redondance des chemins : est une mesure du nombre de chemins alternatifs dans le graphe.</p> $\Psi_t(G) = \frac{1}{2N_G(t)} \sum_{u,v \in V^2} P_{uv} $
<p>cout de routage : Il s'agit du nombre total de paquets à transmettre dans le réseau, y compris les retransmissions multi-sauts, pour que chaque sommet atteigne tous les autres → reflète aussi l'efficacité énergétique</p> $R_t(G) = \sum_{u,v \in V^2} d(u,v)$	<p>disparité de chemins : Elle mesure la différence entre les chemins alternatifs, c'est-à-dire le nombre de sommets distincts composant ces chemins.</p> $\delta(u,v) = \frac{1}{2 P_{uv} } \sum_{p_i, p_j \in P_{uv}^2} \frac{ p_i(u,v) \oplus p_j(u,v) }{d(u,v) - 1} \quad \Delta_t(G) = \frac{1}{2N_G(t)} \sum_{u,v \in V^2} \delta(u,v)$
<p>efficacité du réseau : Cette métrique, comprise entre 0 et 1, caractérise la distance moyenne entre les paires de sommets.</p> $\Theta_t(G) = \frac{1}{2N_G(t)} \sum_{u,v \in V^2, u \neq v} \theta(u,v) \quad \theta(u,v) = \frac{1}{d(u,v)}$	<p>criticité des nœuds : est un indicateur de la présence et du nombre de sommets critiques dans un graphe. Un sommet est considéré comme critique s'il est présent dans une grande partie des plus courts chemins du graphe</p> $BC_t(i) = \frac{1}{2N_G(t)} \sum_{u,v \in V^2} \frac{ P_{uv}(i) }{ P_{uv} }$

Métriques supplémentaires

Équité de la division (RMSE) : mesure l'écart entre les tailles des groupes formés par un algorithme de division. Un RMSE faible indique une répartition équilibrée

6 Méthodologie de Simulation

Le scénario modélise un groupe de **100 nanosatellites** évoluant dans l'espace pendant **1000 unités de temps** (*le max pour avoir une simulation rapide*). À chaque pas de temps t , les coordonnées (x, y, z) de chaque satellite sont extraites depuis "traces.csv".

Les scripts développés suivants ont été utilisés :

- `swarm_sim.py` : création de la topologie du réseau (construction du graphe à partir des positions).
- `metrics.py` : calcul des différentes métriques de robustesse et résilience.
- `simulation.ipynb` : script principal de simulation et de visualisation, automatisant les appels aux fonctions de simulation et d'analyse.

Étapes de la simulation

Pour chaque instant t dans l'intervalle simulé :

1. **Chargement des positions** des satellites à t depuis `Traces.csv`.
2. **Construction du graphe** $G_t = (V_t, E_t)$
3. **Application d'un algorithme de division** : RND, FFD, MIRW ou k-Means.
4. **Calcul des métriques avant panne** :
 - Flow robustness, routing cost, network efficiency, etc.
5. **Simulation d'une panne** à un instant donné (ex. $t = 500$) :
 - Suppression d'un nombre de nœuds (aléatoire ou ciblée),
 - Recalcul des métriques après panne.
6. **Enregistrement des résultats** dans un fichier `.csv` spécifique à chaque méthode :
 - `resultats_swarm_simulation_RND.csv`
 - `resultats_swarm_simulation_FFD.csv`
 - `resultats_swarm_simulation_MIRW.csv`
 - `resultats_swarm_simulation_kMeans.csv`

7 Visualisations

7.1 Visualisation des trajectoires orbitales

Avant d'appliquer les algorithmes de division de graphe ou de simuler des pannes dans l'essaim de nanosatellites, il est indispensable de comprendre visuellement la dynamique spatiale des satellites au cours du temps. Cette étape permet d'évaluer la distribution et le comportement orbital des satellites simulés, qui influencent directement la connectivité du réseau.

Exemple de trajectoire individuelle

La Figure 4 ci-dessous montre la trajectoire du satellite numéro 53 au cours du temps. On observe un mouvement orbital stable et régulier, qui correspond à une orbite circulaire inclinée par rapport au plan équatorial. Cette visualisation confirme que le modèle génère une dynamique cohérente avec un comportement réaliste pour un satellite en orbite basse.

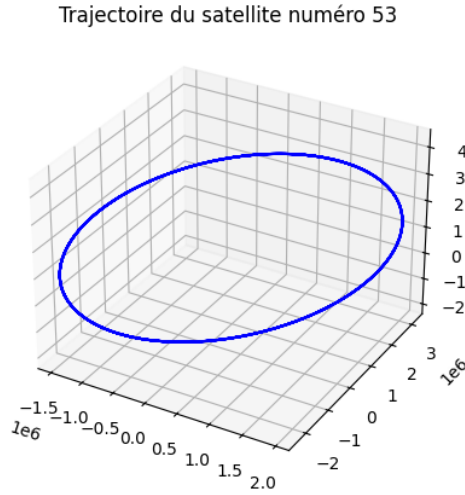


FIGURE 4 – Trajectoire du satellite numéro 53

Vue globale des trajectoires

La Figure 5 suivante illustre les trajectoires de l'ensemble des 100 satellites simulés. Ces trajectoires montrent une distribution relativement homogène dans l'espace, avec des croisements fréquents, ce qui est représentatif d'un essaim dynamique fortement interconnecté.

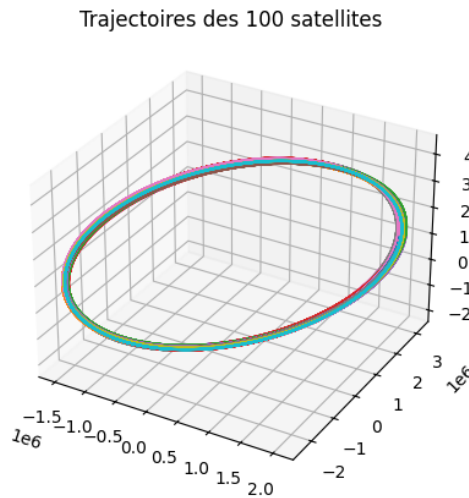


FIGURE 5 – Trajectoires des 100 satellites simulés

7.2 Visualisation des graphes du réseau à un instant donné

Afin d'analyser visuellement la structure du réseau et les effets des différents algorithmes de division, nous avons généré des représentations graphiques du graphe de l'essaim à un instant t . Ces visualisations permettent de mieux comprendre comment les algorithmes influencent la connectivité intra-groupe, la séparation inter-groupes, ainsi que la topologie globale du réseau.

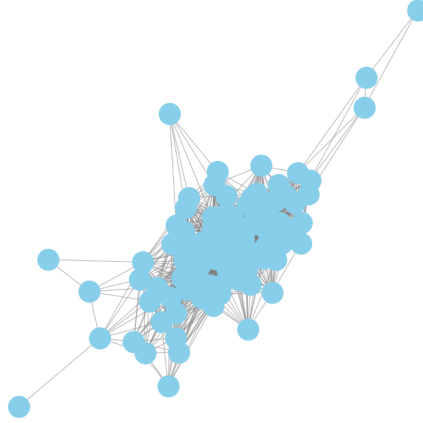


FIGURE 6 – Graphe de l'essaim avant division ($t = 50$)

La figure 6 montre la structure du réseau sans division. Les nœuds sont connectés selon les portées de communication, et tous les satellites sont considérés dans un seul groupe. Cette représentation illustre une forte densité de connexions, rendant le réseau difficile à organiser ou à optimiser sans stratégie de structuration.

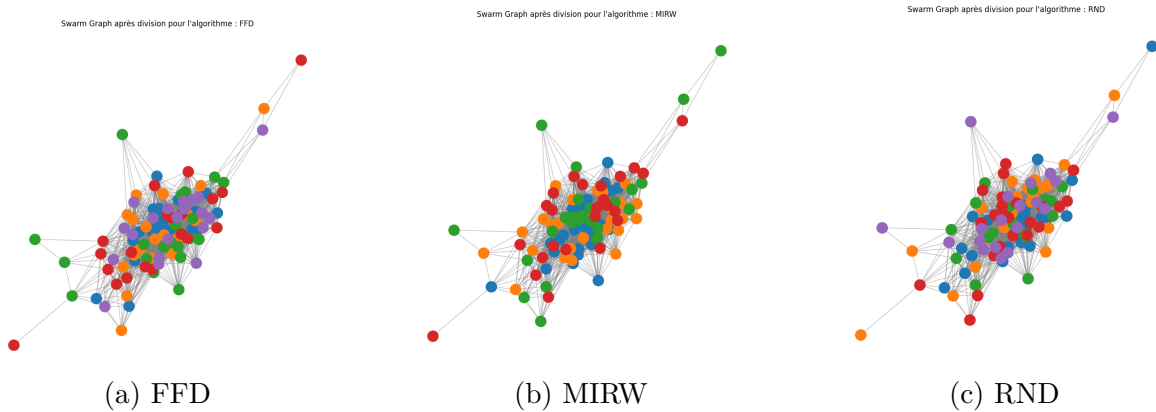


FIGURE 7 – Visualisation des graphes après division pour les trois algorithmes : FFD, MIRW et RND

Dans les figures 7a, 7b et 7c, nous visualisons le graphe après l'application respective de chaque algorithme de division. Ces représentations permettent d'observer :

- une segmentation claire des sous-groupes,

- des zones à forte connectivité locale,
- et des différences significatives dans la répartition et la compacité des clusters générés.

Commentaires :

- L'algorithme **FFD** crée des groupes compacts et bien séparés. On observe une meilleure cohérence topologique intra-groupe.
- L'algorithme **MIRW** crée des groupes légèrement moins denses que FFD, avec des frontières parfois moins nettes.
- L'algorithme **RND**, étant basé sur une répartition aléatoire, donne des groupes beaucoup plus mélangés, avec une faible cohésion structurelle.

Ces visualisations appuient les analyses quantitatives de robustesse, efficacité et disparité. En effet, un bon algorithme de division doit non seulement maximiser la résilience, mais aussi favoriser des regroupements structurés pour faciliter le routage et la tolérance aux pannes.

8 Analyse des résultats

8.1 Contexte expérimental

L'analyse repose sur l'observation des métriques de résilience et de robustesse calculées sur un essaim de 100 nanosatellites simulé sur 1000 instants. Une attaque ciblée est appliquée à $t = 500$, supprimant 10 % des nœuds les plus centraux. Nous comparons cinq configurations :

- Graphe **non divisé**,
- Division selon les algorithmes **FFD**, **MIRW**, **RND**,
- et une méthode supplémentaire : **k-Means**.

8.2 Robustesse :

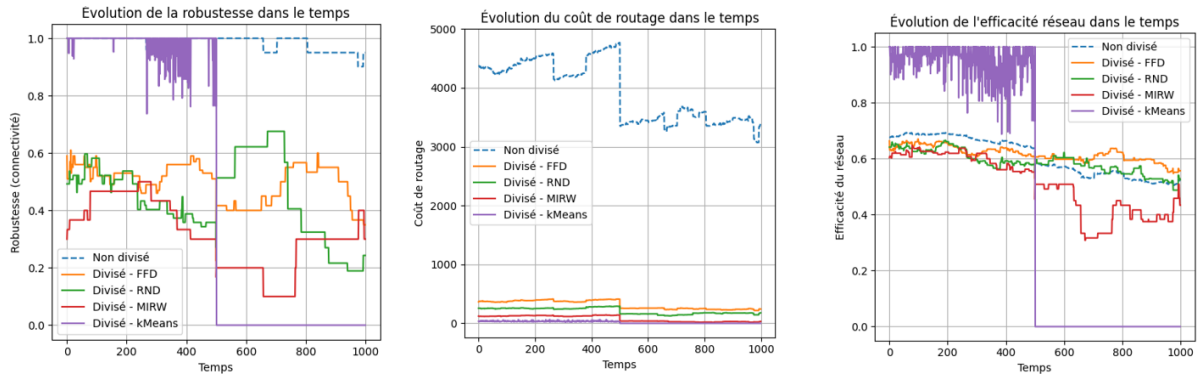


FIGURE 8 – évolution des métriques de robustesse dans le temps

Robustesse (Flow Robustness). Comme le montre la Figure 10 le graphe non divisé maintient une robustesse de flux stable ; malgré la suppression de 10 des nœuds critiques. Cela confirme que le réseau global est structurellement robuste mais vulnérable à la perte de certains nœuds centraux. La baisse de robustesse observée avec les algorithmes de division s'explique par la nature même de la partition : en structurant le réseau en clusters, les communications se concentrent à l'intérieur des groupes, réduisant les liens inter-groupes. Ainsi, la robustesse globale mesurée comme la proportion de paires connectées dans tout le graphe diminue mécaniquement, bien que **la robustesse locale au sein des clusters s'améliore**. Ce compromis est volontaire : *il favorise une meilleure tolérance aux pannes locales, au détriment de la connectivité globale*.

Coût de routage. Le graphe non divisé engendre un coût de routage élevé, dû à la nécessité de maintenir la connectivité entre tous les nœuds. À l'inverse, toutes les méthodes de division réduisent ce coût, en particulier **k-Means** et **FFD**. Cela reflète un *gain d'efficacité énergétique* dans la communication intra-groupe.

Efficacité du réseau. L'efficacité du réseau, inversement proportionnelle à la distance moyenne entre les nœuds, décroît après la panne dans tous les cas. **FFD** et **MIRW** conservent des niveaux acceptables, tandis que **k-Means** s'effondre brutalement, révélant une perte de connexions critiques. Ces observations confirment que la compacité initiale ne garantit pas la résilience structurelle dans un graphe dynamique.

Niveau	Robustesse	Coût de routage	Efficacité réseau
Très bon	Non divisé, k-Means (avant panne)	MIRW, k-Means	FFD, RND
Bon	FFD, MIRW	RND	MIRW
Moyen	RND	FFD	Non divisé
Faible	k-Means (après), RND (post-panne)	Non divisé	k-Means –

TABLE 1 – Classification des algorithmes selon trois métriques de robustesse

8.3 Résilience :

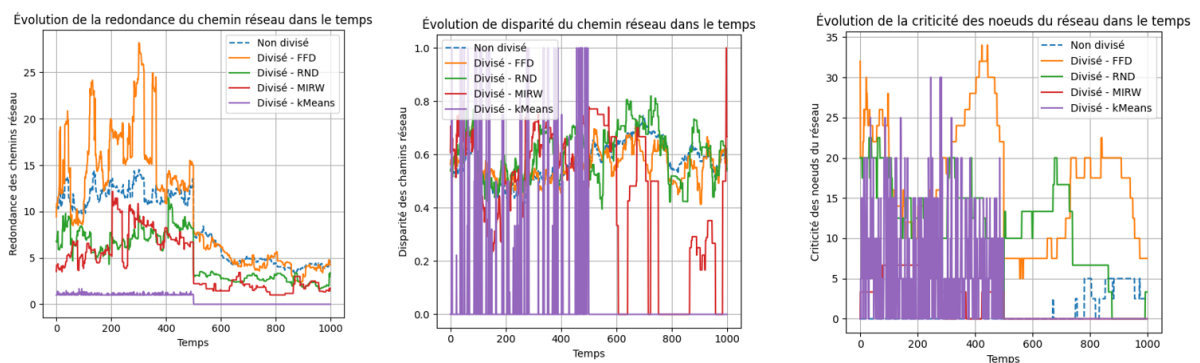


FIGURE 9 – évolution des métriques de résilience dans le temps

Redondance des chemins. La redondance est essentielle pour la tolérance aux pannes, car elle garantit la présence de chemins alternatifs. **FFD** se démarque clairement avec une forte redondance (jusqu'à 25 chemins alternatifs), ce qui démontre une *résilience topologique élevée* (Figure 11.a). **MIRW** est modéré, tandis que **RND** et **k-Means** offrent une redondance faible voire nulle.

Disparité des chemins. La disparité reflète la diversité topologique entre les chemins. Une forte disparité indique que les chemins ne passent pas tous par les mêmes sommets, réduisant la vulnérabilité à une panne unique. **FFD** et **MIRW** affichent de bons niveaux de disparité (Figure 11.b), traduisant une structuration équilibrée du graphe.

Criticité des nœuds. L'analyse de la criticité (Figure 11.c) révèle que la division du graphe peut concentrer le trafic sur certains nœuds, les rendant critiques. C'est notamment le cas de **FFD**, qui montre des sommets très centraux. **MIRW** et **k-Means** produisent une criticité plus diffuse. Le graphe non divisé présente une faible criticité individuelle, mais au prix d'un coût élevé en redondance et en énergie.

Niveau	Redondance des chemins	Disparité des chemins	Criticité des nœuds
Très bon	FFD	FFD, MIRW	Non divisé
Bon	RND	RND	MIRW
Moyen	MIRW	Non divisé	k-Means
Faible	k-Means	k-Means	FFD, RND

TABLE 2 – Classification des algorithmes selon les métriques de résilience

8.4 Analyse statistique globale

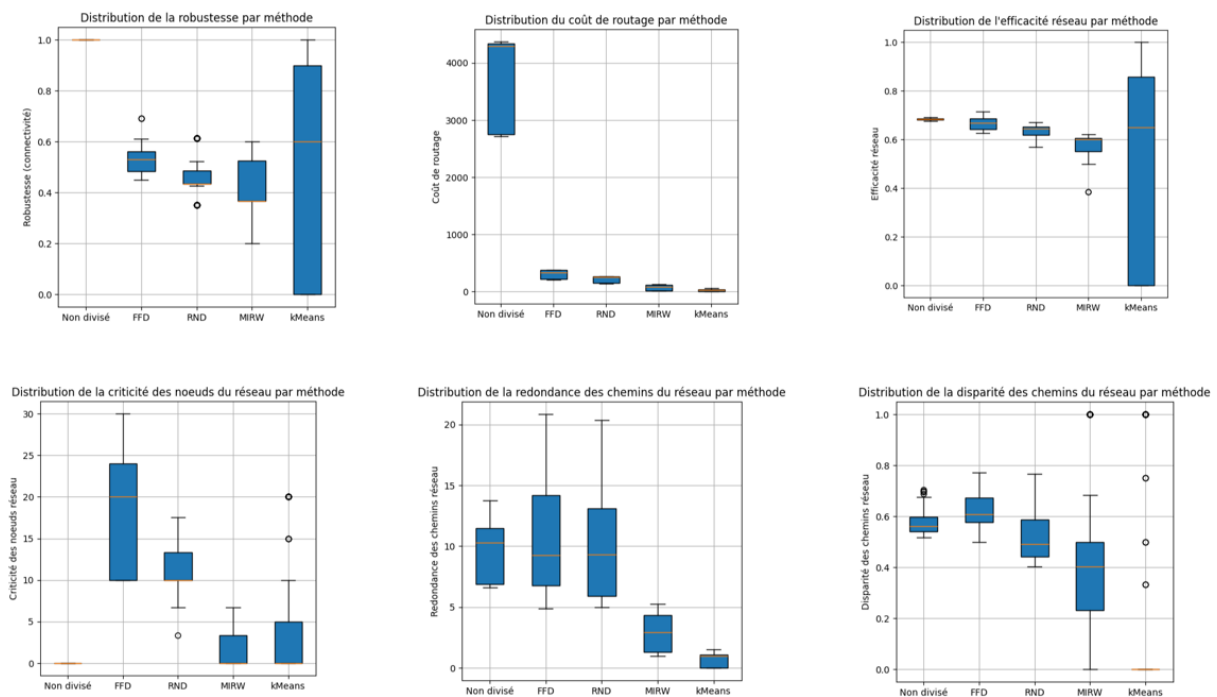


FIGURE 10 – distributions des métriques de robustesse et de résilience par méthode de division

Les distributions (Figure 12) confirment les tendances précédentes :

- **FFD** offre une excellente résilience (redondance, disparité), au prix d'une criticité plus élevée.
- **MIRW** est le meilleur compromis pour des clusters équilibrés, efficaces et tolérants aux pannes.

- **RND** est instable et sensible aux attaques ciblées.
- **k-Means** est très variable et peu fiable sans prise en compte topologique.

8.5 Équité des groupes : RMSE

L'objectif de cette analyse est d'évaluer l'équité de la répartition des nœuds entre les groupes à l'aide de la métrique RMSE (Root Mean Square Error) appliquée aux tailles des groupes. Plus le RMSE est bas, plus la répartition est homogène, ce qui est souhaitable pour garantir une charge équilibrée et une gestion équitable des ressources.

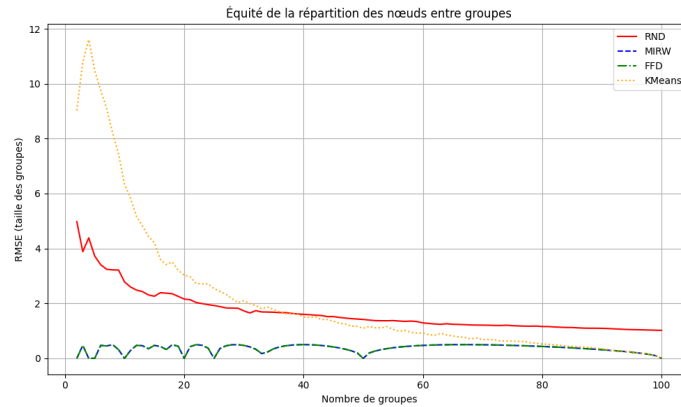


FIGURE 11 – Tracé de RMSE en fonction du nombre de groupes

Algorithme	Equite (RMSE)	Commentaire
RND	Moyenne	Diminue avec nb groupes ; correct si simplicité priorisée
MIRW	Excellente	Tres stable ; idéal pour équilibre entre groupes
FFD	Excellente	Tres stable ; même performance que MIRW
KMeans	Mauvaise au debut	S'améliore, mais tres déséquilibre pour petit k

FIGURE 12 – Résultats synthétiques de RMSE par algorithme

En conclusion, **MIRW** et **FFD** sont les plus efficaces pour garantir une répartition équitable des nœuds entre groupes. Ce sont les algorithmes les plus adaptés lorsqu'un bon équilibre est un critère prioritaire.

9 Conclusion

Niveau	Robustesse	Routing	Efficacité	Redondance	Disparité	Criticité
Très bon	Nondivisé	MIRW,k-Means	FFD,RND	FFD	FFD,MIRW	Nondivisé
Bon	FFD,MIRW	RND	MIRW	RND	RND	MIRW
Moyen	RND	FFD	Nondivisé	MIRW	Nondivisé	k-Means
Faible	k-Means,RND	Nondivisé	k-Means	k-Means	k-Means	RND,FFD

TABLE 3 – Classification des algorithmes selon les principales métriques réseau

Niveau	RMSE (Équité)	Complexité algorithmique
Très bon	FFD, MIRW	k-Means
Bon	RND	RND
Moyen	Non divisé	FFD
Faible	k-Means (petit k)	MIRW

TABLE 4 – Classification des algorithmes selon l'équité de répartition (RMSE) et leur complexité

Les résultats montrent que les algorithmes structurés comme **FFD** et **MIRW** offrent un bon équilibre entre **résilience locale et efficacité de communication**, en conservant des clusters cohérents et redondants même après panne. **MIRW**, en particulier, présente une excellente équité de la division (faible RMSE), ce qui en fait un candidat robuste pour des déploiements équilibrés. À l'inverse, **RND**, bien que simple, souffre d'une instabilité structurelle et d'une variabilité forte. Quant à **k-Means**, il montre un bon comportement global lorsque le nombre de clusters est élevé, mais s'avère très déséquilibré pour des petites partitions (RMSE élevé), limitant son usage dans des configurations réduites.

u-delà de son rôle structurel, **la division de graphe s'est révélée être un outil analytique puissant pour l'étude de la résilience d'un essaim de nanosatellites**. En partitionnant le réseau, nous avons pu mieux observer et quantifier *l'impact localisé des pannes*, identifier *les groupes vulnérables*, et mettre en évidence *les zones critiques*, notamment les nœuds fortement centraux ou les liens inter-groupes sensibles.

Cette approche permet non seulement de comparer différentes stratégies de découpage (FFD, MIRW, etc.) selon des critères de performance comme la robustesse, l'efficacité ou l'équité (RMSE), mais aussi de **mieux comprendre la topologie fonctionnelle du réseau**. En simulant des attaques ciblées et en suivant l'évolution des métriques intra- et inter-groupes, la division offre une **lecture fine des faiblesses structurelles potentielles du graphe**.

Ainsi, la division de graphe n'est pas seulement une technique d'optimisation ou de structuration : elle devient un levier d'analyse stratégique pour anticiper les défaillances, localiser les points de rupture possibles, et concevoir des architectures de communication plus robustes et tolérantes aux pannes.

Pistes d'amélioration

- **Optimisation multi-objectifs** : Un futur axe serait d'optimiser conjointement la robustesse, la redondance et la criticité, via des approches heuristiques ou d'apprentissage.
- **Scalabilité et simulation à grande échelle** : Étendre les simulations à des essaims de plus de 100 nœuds permettrait de tester la robustesse algorithmique face à une montée en échelle, en particulier pour les méthodes à complexité non linéaire.

Références

- [1] Evelyn Akopyan. *Analyse de la résilience topologique dans les essaims de nanosatellites*. Thèse de doctorat, École polytechnique, 2024.
- [2] Amine Hanafi. *Étude de la fiabilité et de la longévité du nanosatellite universitaire MASAT-1*. Mémoire de master, Université de Budapest, 2022.
- [3] Sarah Meziane-Tani. *Étude de faisabilité d'une constellation de nanosatellites Cubesat pour la collecte de données à distance*. Mémoire de master, École Nationale Supérieure d'Informatique (ESI), 2018.