Network engineering

Clustering in nanosatellite swarms

REY, POINTEAU, BAITA, DREUMONT, POPESCU-CRISTEA

Introduction

Nanosatellite swarm characteristics

-Low Earth Orbit(LEO)
-Dynamic topology
-Resource constraints
-Need Scalability
-Need Fiability

State of the art

The swarms of nanosatellites are groups made up of small, low-cost satellites, which are increasingly used for applications such as Earth observation and communication.

Clustering Algorithms

Weight-based Dominating Set Clustering

This algorithm selects cluster heads based on weights assigned to each nanosatellite, considering factors like energy and connectivity. It ensures efficient and stable group formation, particularly well-suited for varying satellite densities and sizes.

Agent-based Clustering Framework

Introduced in 2021, this approach models satellites as autonomous agents within a hierarchical structure, featuring three control levels for component management. It uses a new efficiency indicator to ensure stability and scalability, addressing the critical resource constraints of nanosatellite swarms.

Virtual Agent Clustering (VAC)

This approach introduces virtual agent clusters to manage mobility, aiming to reduce overhead and packet switching delays in LEO satellite networks. It's especially relevant for nanosatellites, where high mobility and frequent transfers can strain resources.

Routing Algorithms

Open Shortest Path First (OSPF)

OSPF is a widely used routing protocol designed for static networks. However, its effectiveness is limited in dynamic environments like nanosatellite swarms. Consequently, researchers have explored adaptations of OSPF to better suit dynamic conditions.

Delay-Tolerant Network (DTN)

DTN is a protocol that effectively handles high latency, making it well-suited for nanosatellite swarms. Researchers have demonstrated its viability for space communications, and an extension called Deep Space Network (DSN) routing has also been applied to nanosatellite scenarios.

Simulations

Simulations

Graph class

Animation class

Sim class

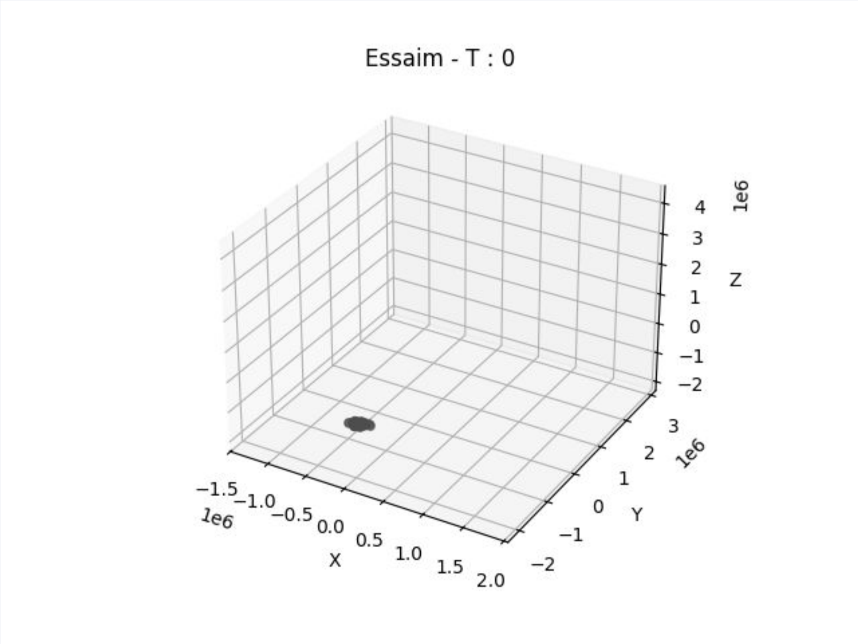
Swarm dynamics

Data analysis of the available data :

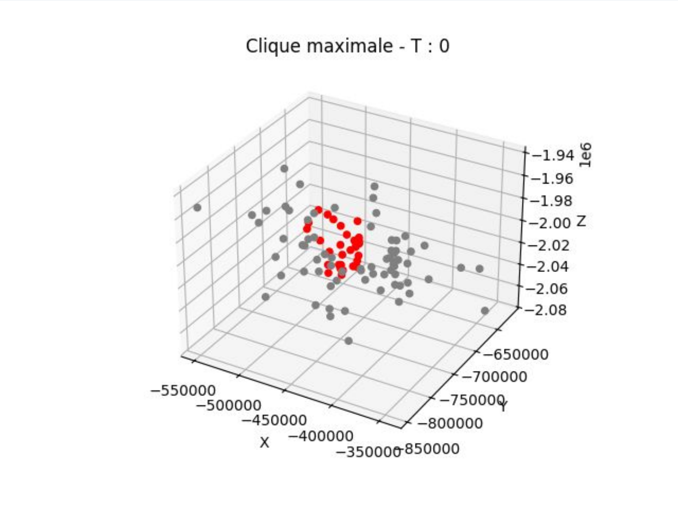
- CSV of positions of 100 satellites
    - x, y and z coordinates
    - over 10000 frames

- One cluster of nanosatellites

- located on several orbits



Swarm dynamics



Some stats about the data :

- 264,113 cliques on average
- Biggest clique of size 32
- Smallest clique of size 3

- 2,272 connected components on average
- 98,431 on average for the size of the maximum connected component
- 0,975 isolated nanosatellite on average
- Up to 5 connected components at max

Clustering

Two algorithms

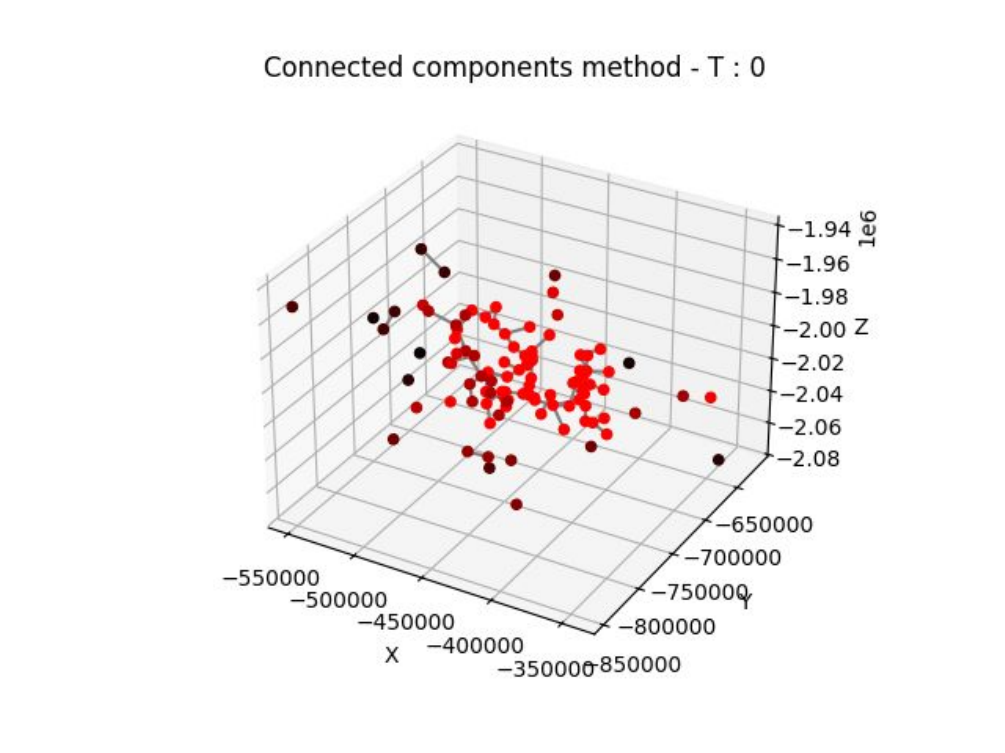
1st algorithm:

Creates clusters from connected components that contains nanosatellites that are at a maximum distance of 20,000km between them.

2nd algorithm:

Apply KMeans to the graph and then take a nanosatellite per cluster to create a cluster and repeat until there is no nanosatellite remaining. Like a random clustering but guided.

Animation - Connected components method



Results - Connected components method

Average power in the swarm: 12585.70

Average throughput in the swarm: 685.18 Mb/s

Average number of isolated nanosatellites: 24.057

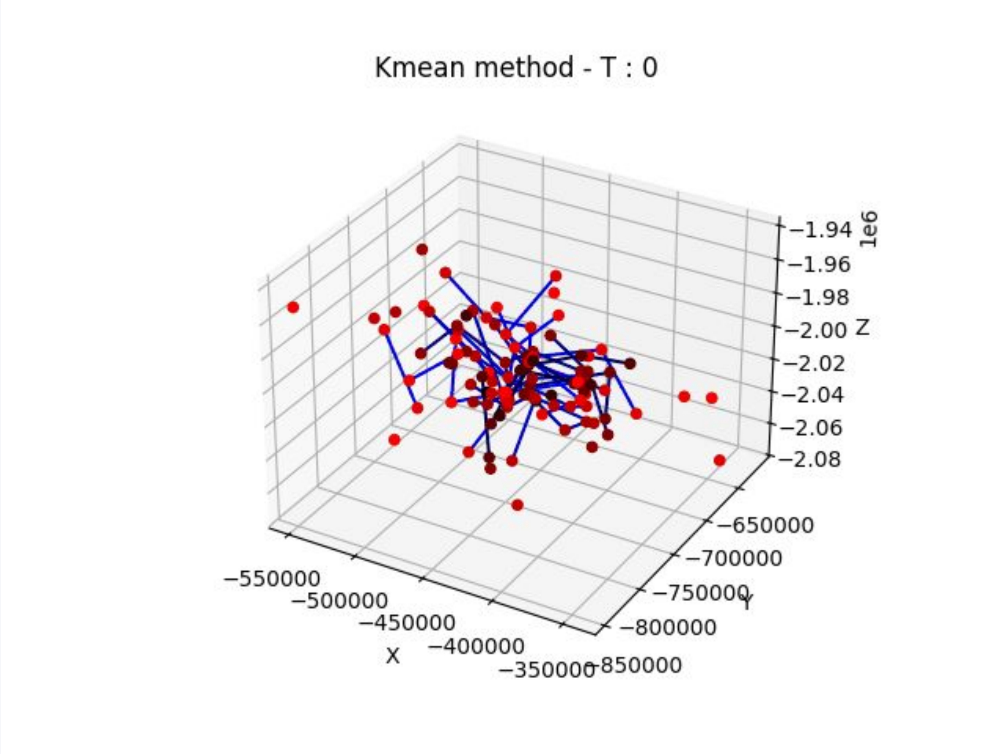
Average number of clusters: 31.482

Average size of clusters: 3.37

Minimum size of clusters: 1

Maximum size of clusters: 83

Animation - KMeans based method



Results - KMeans based method

Average power in the swarm: 105037.8

Average throughput in the swarm: 264.408 Mb/s

Average number of isolated nanosatellites: 27.408

Average number of clusters: 20.06

Average size of clusters: 5.08

Minimum size of clusters: 1

Maximum size of clusters: 10

Better simulation

New technique to link nodes.

- update nodes so that they know their ranges
- Calculate the average power and throughput accordingly

Results - Connected components method

Average power in the swarm: 116982.4

Average throughput in the swarm: 855.658 Mb/s

Average number of isolated nanosatellites: 24.057

Average number of clusters: 31.482

Average size of clusters: 3.37

Minimum size of clusters: 1

Maximum size of clusters: 83

Results - KMeans based method

Average power in the swarm: 287289.2

Average throughput in the swarm: 481.164 Mb/s

Average number of isolated nanosatellites: 27.296

Average number of clusters: 20.129

Average size of clusters: 5.07

Minimum size of clusters: 1

Maximum size of clusters: 10

Conclusion

- Better algorithm: connected component
- We can't really compare our algorithms with already existing ones (algorithms from the state of the art are really complex)

Thanks !

Questions ?