1. Introduction

Dans le cadre de ce projet, nous avons choisi d'appliquer l'algorithme de Floyd-Warshall à l'exemple suivant:
le routage des communications au sein d'une constellation de satellites en orbite basse (LEO).

Les constellations LEO (Low Earth Orbit), comme Starlink, OneWeb ou Kuiper, reposent sur des centaines voire des milliers de satellites orbitant à une altitude comprise entre 500 et 1200 km. Ces satellites servent à couvrir le réseau Internet à un niveau mondial, plus particulièrement dans les zones isolées où les infrastructures terrestres sont limitées.

Pour garantir un service Internet continu, les satellites doivent échanger des données entre eux à travers des **liaisons inter-satellites** (radio ou laser).
Chaque liaison possède un **temps de propagation** (latence), qui dépend de la distance et des conditions orbitales.

Le problème étudié est donc le suivant :

> ***Comment déterminer les chemins de communication de latence minimale entre tous les couples de satellites d'une constellation ?***

On utilise alors l'algorithme de Floyd-Warshall pour calculer, en une seule exécution, les plus courts chemins entre tous les couples de sommets d'un graphe orienté valué.

Avant ça un petit rappel théorique de l'algorithme de Flo yd–Warshall

L'algorithme de Floyd–Warshall est un algorithme matriciel qui permet de calculer comme dit au-dessus, en une seule exécution, les plus courts chemins entre tous les couples de sommets d'un graphe orienté (positif ou négatif, mais sans cycle absorbant (sinon on se retrouve avec des chemins minimums infinis)).

L'algorithme s'appuie sur un principe de programmation dynamique : à l'étape *k*, on suppose que l'on connaît déjà les plus courts chemins n'utilisant que des sommets intermédiaires dans {0..k−1}.

Pour chaque paire (i, j)(i le sommet de départ, et j le sommet d'arrivé), l'algorithme teste donc si passer par le sommet *k* (un sommet intermédiaire à i et j) permet de réduire la distance :

L[i] [j] = min(L[i] [j], L[i] [k] + L[k] [j])

avec

- **L** est la matrice des coûts,

- **P** est la matrice des prédécesseurs,
- **L[i][j] = INF** signifie qu'il n'existe pas d'arête directe entre i et j.

Initialisation :

- L[i][j] = poids(i→j) si l'arête existe
- L[i][j] = INF sinon
- P[i][j] = i si l'arête existe, -1 sinon
- L[i][i] = 0

Algorithme (pseudo-code) :

```
pour k = 0 à n−1 :
   pour i = 0 à n−1 :
      pour j = 0 à n−1 :
         si L[i] [k] + L[k] [j] < L[i] [j] :
            L[i] [j] = L[i] [k] + L[k] [j]
            P[i] [j] = P[k] [j]
                  fin
         fin
fin
```

2. Modélisation du réseau de satellites

Nous avons modélisé une constellation simplifiée de 15 satellites, chacun représentant une zone géographique du globe.
Chaque satellite correspond à un sommet du graphe, numéroté de 0 à 14.
Les arcs représentent les liaisons inter-satellites directement accessibles et les poids sont des latences moyennes en millisecondes.

Correspondance satellites → zones géographiques

Satellite	Zone représentée
0	Ouest des États-Unis
1	Centre des États-Unis
2	Est des États-Unis
3	Atlantique Nord

4	Europe du Nord
5	Europe du Sud
6	Afrique du Nord
7	Moyen-Orient
8	Inde
9	Asie du Sud-Est
10	Chine
11	Japon
12	Pacifique Nord
13	Océanie / Pacifique Sud
14	Amérique du Sud Est

Les liaisons sont symétriques (un arc dans les deux sens), afin de modéliser des communications bidirectionnelles.
Les latences sont comprises entre 20 et 95 ms (valeurs réalistes pour des liaisons optiques inter-satellites).

3. Données & matrice des poids

Nous avons construit un graphe de :

- **n = 15 sommets**
- **m = 62 arcs**

Les données sont dans le fichier :
“**tests/satellites15.txt”**

Le tableau suivant présente un extrait de la matrice des poids :

```
Nom du fichier (ou 'q' pour quitter) : tests/satellites15.txt

Graphe chargé depuis 'tests/satellites15.txt'
Matrice des poids :
          0      1      2      3      4      5      6      7      8      9     10     11     12     13     14
     ------------------------------------------------------------------------------------------------------------
  0 |     0     25    INF     40    INF    INF    INF    INF    INF    INF    INF    INF     70    INF    INF
  1 |    25      0     20    INF     50    INF    INF    INF    INF    INF    INF    INF    INF    INF    INF
  2 |   INF     20      0     25     60    INF    INF    INF    INF    INF    INF    INF    INF    INF     95
  3 |    40    INF     25      0     35    INF    INF    INF    INF    INF    INF    INF     80    INF    INF
  4 |   INF     50     60     35      0     30     45    INF    INF    INF     90    INF    INF    INF    INF
  5 |   INF    INF    INF    INF     30      0     25     55    INF     85    INF    INF    INF    INF    INF
  6 |   INF    INF    INF    INF     45     25      0     35    INF    INF    INF    INF    INF    INF     75
  7 |   INF    INF    INF    INF    INF     55     35      0     30     45    INF    INF    INF    INF    INF
  8 |   INF    INF    INF    INF    INF    INF    INF     30      0     20     35    INF    INF    INF    INF
  9 |   INF    INF    INF    INF    INF     85    INF     45     20      0     25     40    INF     50    INF
 10 |   INF    INF    INF    INF     90    INF    INF    INF     35     25      0     20     60    INF    INF
 11 |   INF    INF    INF    INF    INF    INF    INF    INF    INF     40     20      0     45     70    INF
 12 |    70    INF    INF     80    INF    INF    INF    INF    INF    INF     60     45      0     40    INF
 13 |   INF    INF    INF    INF    INF    INF    INF    INF    INF     50    INF     70     40      0     65
 14 |   INF    INF     95    INF    INF    INF     75    INF    INF    INF    INF    INF    INF     65      0
```

Cette matrice montre les latences directes entre satellites voisins, et s'il n'y a pas de liaisons ont remplit avec "`INF`".

4. Résultats de Floyd-Warshall et interprétations

Après exécution du programme sur ces données, l'algorithme produit :

- les chemins optimaux en termes de latence ;
- la garantie qu'aucun cycle absorbant n'est présent (valeurs strictement positives).
- la matrice des distances minimales entre tous les couples de satellites ;

```
Distances minimales :
          0      1      2      3      4      5      6      7      8      9     10     11     12     13     14
     ------------------------------------------------------------------------------------------------------------
  0 |     0     25     45     40     75    105    120    155    165    155    130    115     70    110    140
  1 |    25      0     20     45     50     80     95    130    160    165    140    140     95    135    115
  2 |    45     20      0     25     60     90    105    140    170    175    150    150    105    145     95
  3 |    40     45     25      0     35     65     80    115    145    150    125    125     80    120    120
  4 |    75     50     60     35      0     30     45     80    110    115     90    110    115    155    120
  5 |   105     80     90     65     30      0     25     55     85     85    110    125    145    135    100
  6 |   120     95    105     80     45     25      0     35     65     80    100    120    160    130     75
  7 |   155    130    140    115     80     55     35      0     30     45     65     85    125     95    110
  8 |   165    160    170    145    110     85     65     30      0     20     35     55     95     70    135
  9 |   155    165    175    150    115     85     80     45     20      0     25     40     85     50    115
 10 |   130    140    150    125     90    110    100     65     35     25      0     20     60     75    140
 11 |   115    140    150    125    110    125    120     85     55     40     20      0     45     70    135
 12 |    70     95    105     80    115    145    160    125     95     85     60     45      0     40    105
 13 |   110    135    145    120    155    135    130     95     70     50     75     70     40      0     65
 14 |   140    115     95    120    120    100     75    110    135    115    140    135    105     65      0
```

4.1 Exemple : plus court chemin États-Unis → Japon

Satellite 0 → Satellite 11

Chemin obtenu :

```
Afficher un plus court chemin ? (o/n) o
Sommet de départ : 1
Sommet d'arrivée : 11

Coût minimal = 140
Chemin : 1 -> 0 -> 12 -> 11

Afficher un plus court chemin ? (o/n)
```

Le chemin montre que le routage optimal passe par l’Atlantique Nord et l’Europe plutôt que par le Pacifique, car les liens y sont plus courts dans la modélisation.
Floyd-Warshall montre donc un itinéraire qui serait impossible à deviner à l’œil nu.

4.2 Exemple : plus court chemin Europe du Nord → Océanie

Satellite 4 → Satellite 13

```
Afficher un plus court chemin ? (o/n) o
Sommet de départ : 4
Sommet d'arrivée : 13

Coût minimal = 155
Chemin : 4 -> 3 -> 12 -> 13
```

Ce trajet traverse plusieurs régions intermédiaires. La liaison finale entre le Pacifique Nord (12) et l’Océanie (13) constitue un point de passage critique.

4.3 Analyse des graphes

On voit bien :

- les satellites 4 (Europe du Nord) et 9 (Asie du Sud-Est) qui sont des nœuds centraux, on voit bien qu’ils minimisent les distances moyennes vers les autres.

- les arcs (12 → 13) et (10 → 11) qui eux sont des passages obligés pour plusieurs trajets longue distance.

- Aucune boucle négative.

5. Conclusion

Cette analyse nous montre ainsi que l’algorithme de Floyd-Warshall est efficace et permet l’analyse de réseaux comme une constellation de satellites.

Il permet de calculer les plus courts chemins entre toutes les paires de nœuds, d’identifier les routes optimales et de mettre en évidence le rôle des satellites critiques dans la performance globale.

Même si nous avons utilisé un modèle simplifié (latences fixes et réseau réduit), on peut facilement voir l’intérêt de Floyd-Warshall pour comprendre la structure d’un réseau spatial et optimiser ses communications. On peut également voir comment il permettrait de rendre plus facile le travail d’optimisation des ingénieurs ou responsables de communication.