

Interference-Aware Beam Hopping Scheduling for LEO Satellite Networks using Soft- Conflict Graph Coloring

Jihane Lghoury, Guilherme Iecker Ricardo, Sebastian Euler,
Xiaotian Fu, Ioannis Xirouchakis, Riadh Dhaou

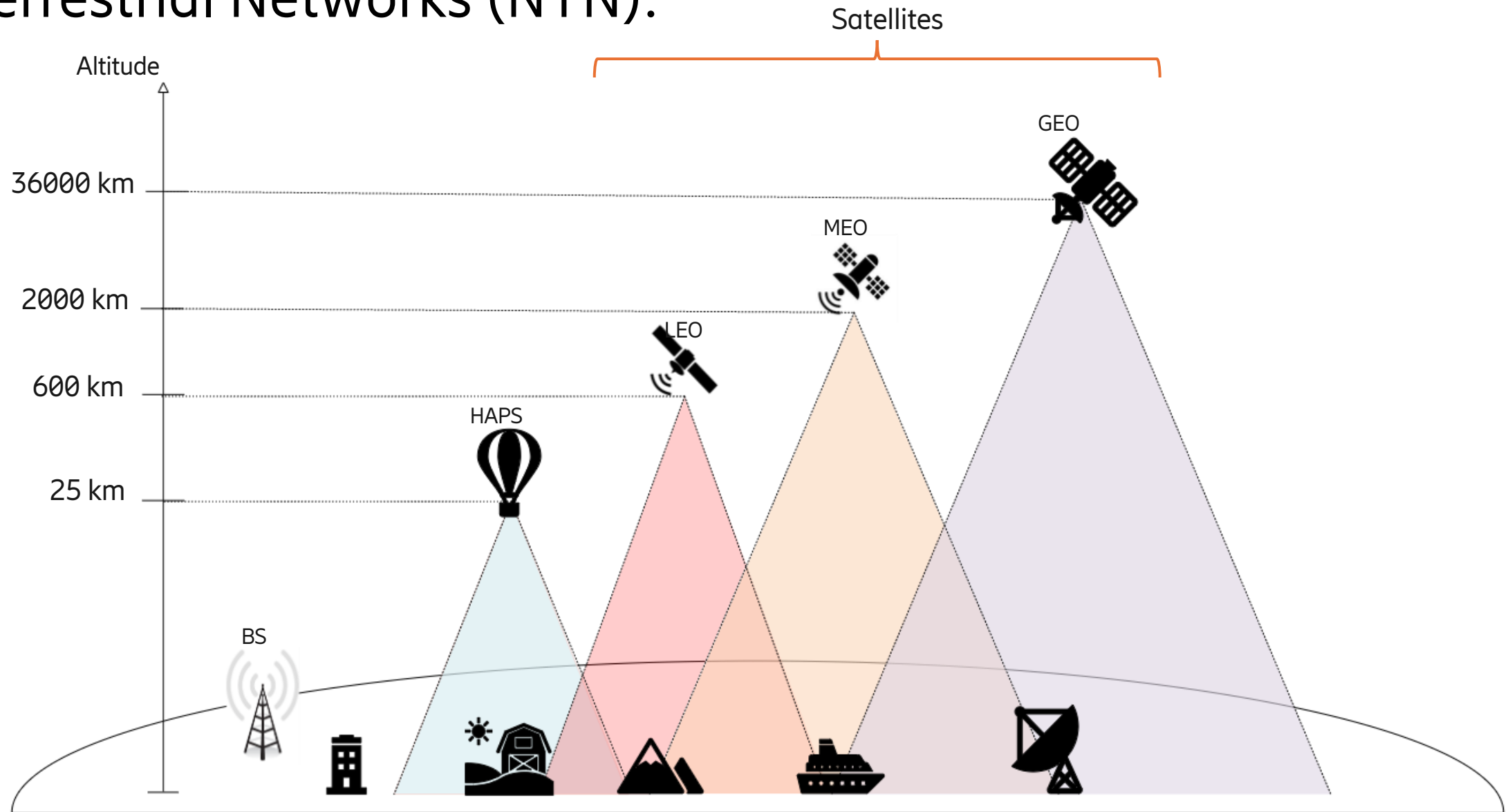


Agenda

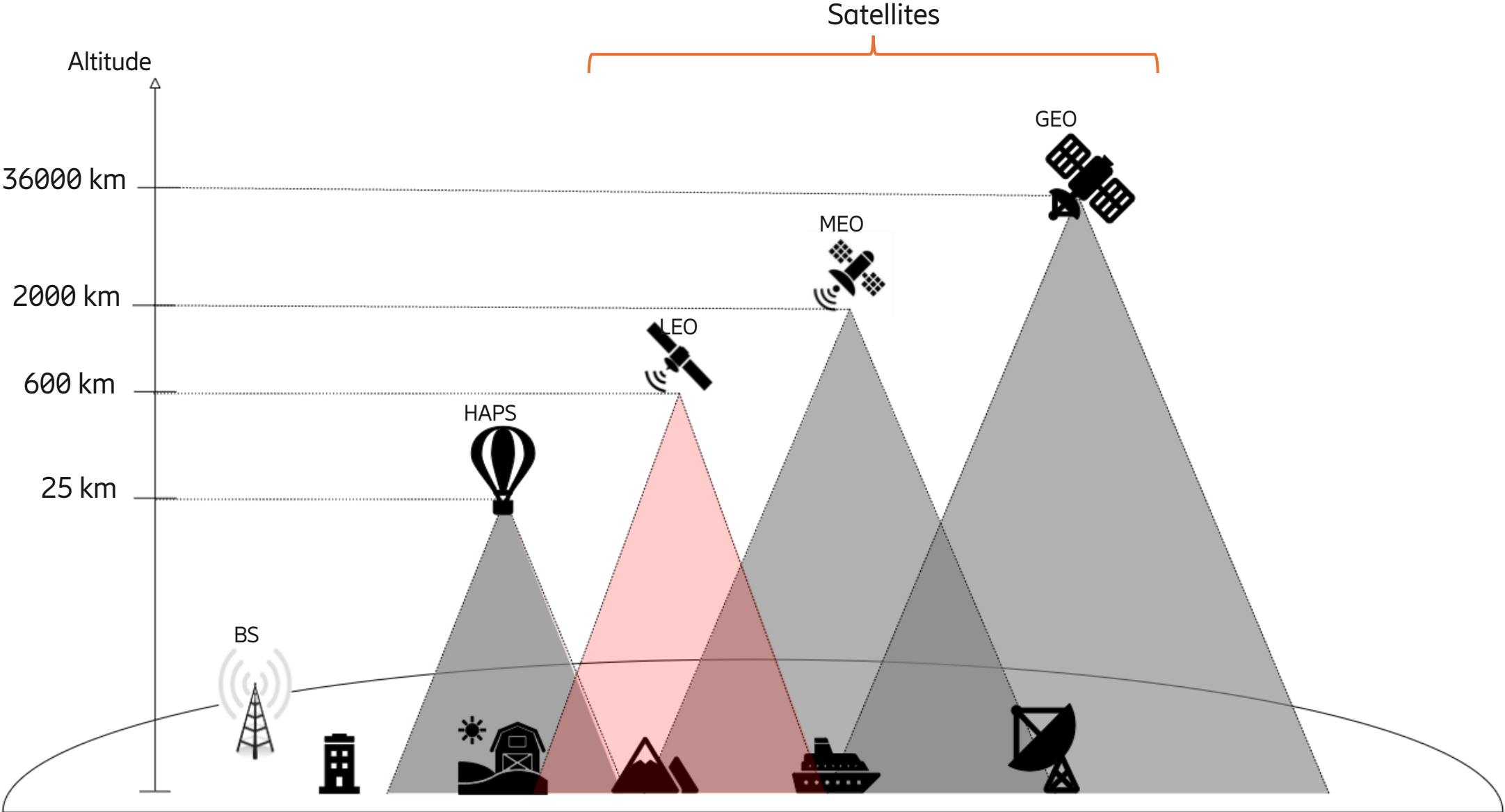
- 01 Context
- 02 System Model
- 03 Graph Coloring- based beam scheduling optimization problem
- 04 Experimental Setup
- 05 Numerical Results

Context

Non-terrestrial Networks (NTN):

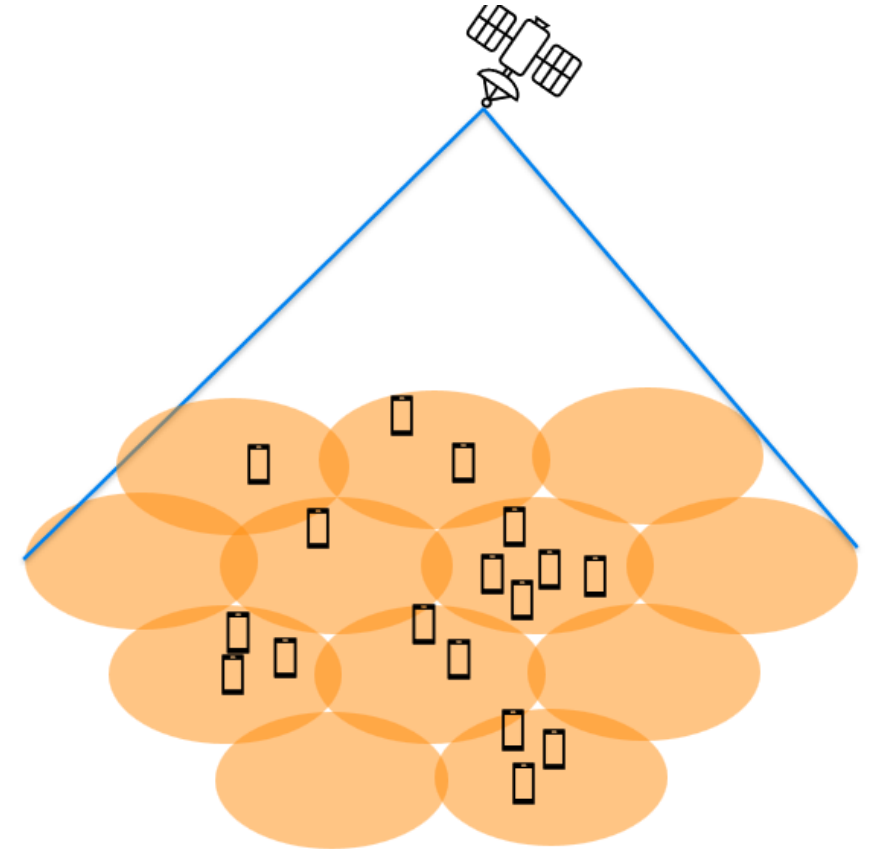
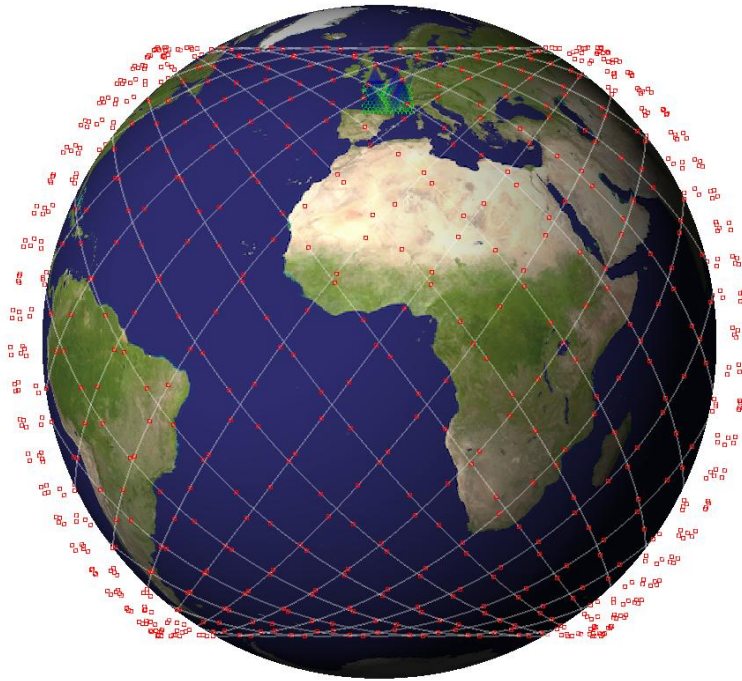


Non-terrestrial Networks (NTN):



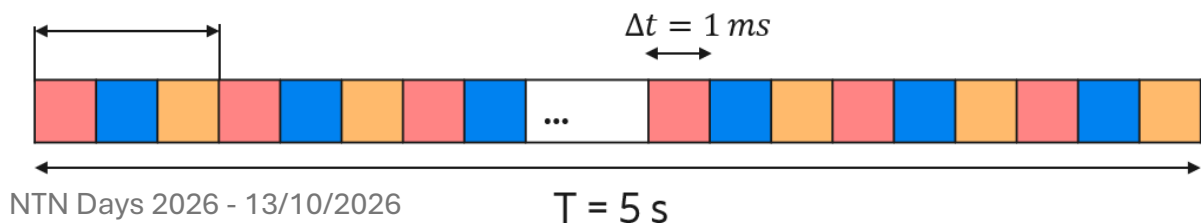
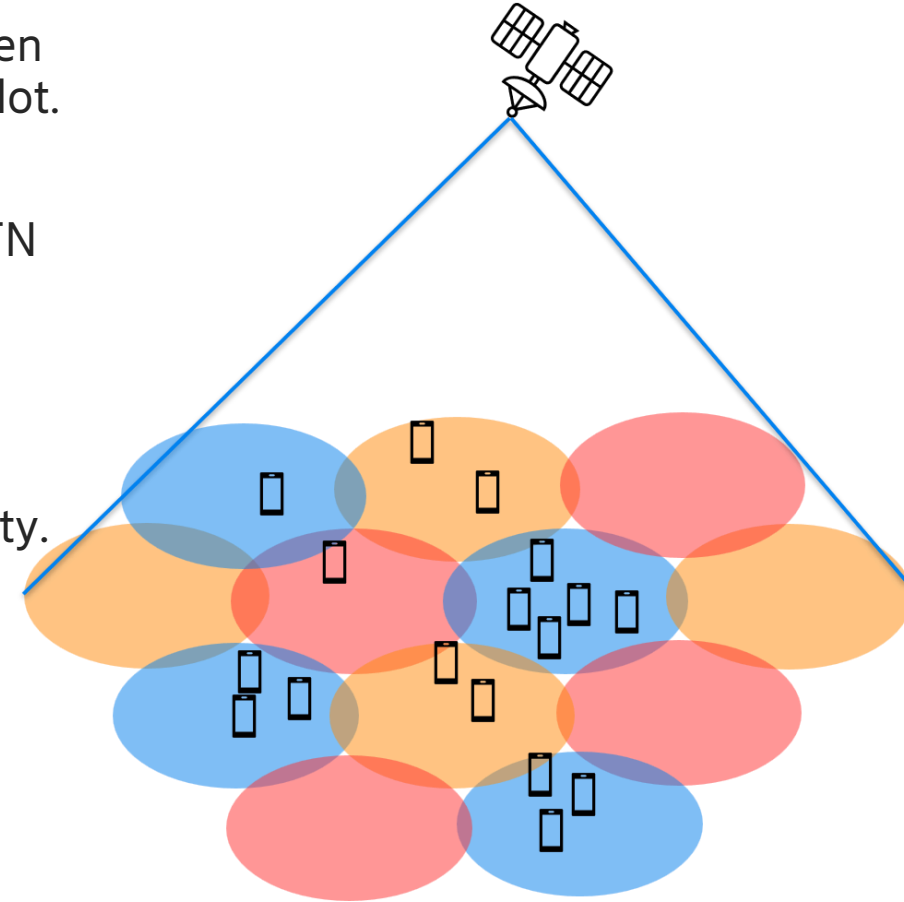
Context

- LEO satellite constellations deploy a large number of satellites to ensure global coverage.
 - LEO multibeam satellites illuminate many ground cells at the same time.
- Simultaneous beams create intra- and inter-satellite interference.



Context

- Beam hopping : a satellite activates only a subset of beams during a given time slot and then switches ("hops") to another subset in the next time slot.
 - Conventional Beam Hopping strategies :
 - 1) often based on fixed geometrical patterns => not well suited to NTN environments : interference is more complex due to 3D geometry, satellite motion, and antenna radiation patterns, resulting in heterogeneous and asymmetric interference relationships.
 - 2) Beam Hopping improves SINR, but time-sharing can reduce capacity.
- How can we improve on conventional methods ?

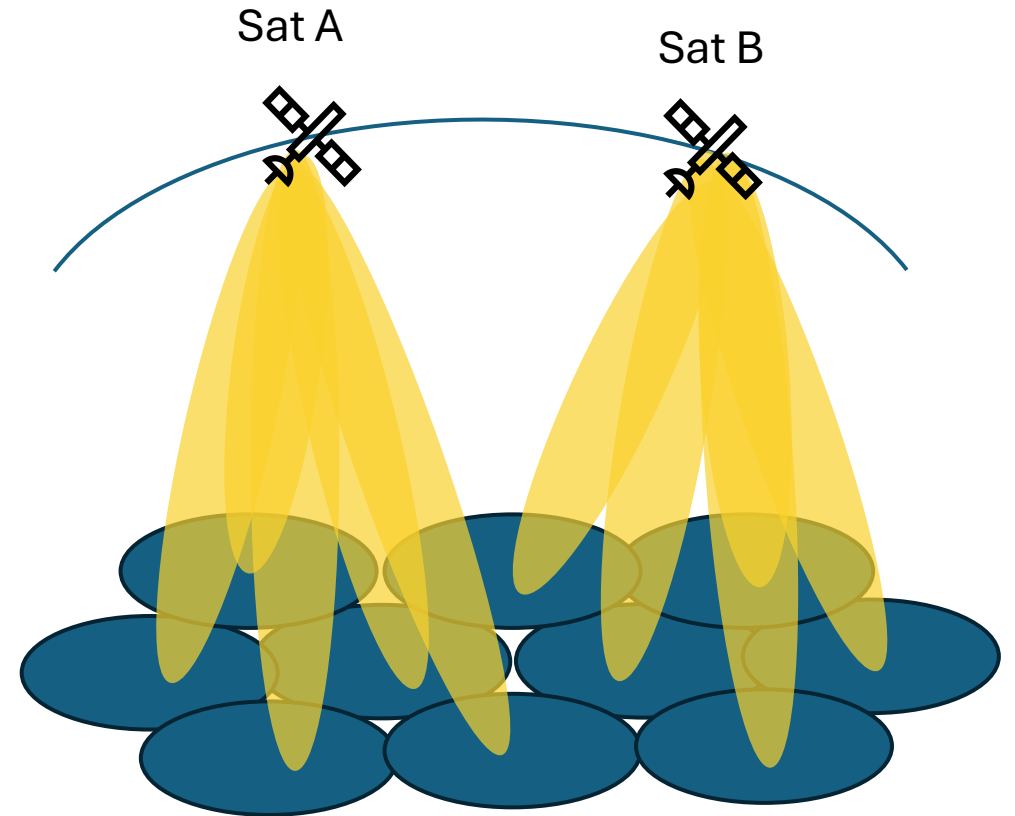


Related work

References	Limitations
[1] : Yaxin Wang et al. "A Flexible Resource Allocation Algorithm in Full Bandwidth Beam Hopping Satellite Systems". In: 2019 IEEE 3rd Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC). 2019 [2] : Zhiyuan Lin et al. "Multi-Satellite Beam Hopping Based on Load Balancing and Interference Avoidance for NGSO Satellite Communication Systems". In: IEEE Transactions on Communications 71.1 (2023)	Interference is approximated using spatial separation or geometric patterns, which does not capture heterogeneous and asymmetric NTN interference.
[3] : Ruishuo Mu et al. "Dynamic Beam Hopping for LEO Satellites with Differentiated Traffic Demands". In: WSCP2023. 2023, pp. 621–626. DOI: 10.1109/WCSP58612.2023.10405252 [4] : Huimin Deng et al. "Satellites Beam Hopping Scheduling for Interference Avoidance". In: IEEE Journal on Selected Areas in Communications 42.12 (2024),	Interference is considered in optimization objectives, but does not explicitly structure the scheduling decisions.
[5] : Zijun Liu et al. "Multi-Satellite Coordinated Beam Hopping for Interference Mitigation Under Tilted Beam Effects: A Graph-Theoretic Approach". In: IEEE Wireless Communications Letters 15 (2026), pp. 2313–2317. DOI: 10.1109/LWC.2026.3676112.	Beam Scheduling based on dynamic satellite–cell assignment and graph coloring.

System model

- Multi-beam satellites with onboard base stations.
- UEs are distributed across a service area, which is partitioned into a set $C = \{1, \dots, C\}$ of fixed cells of equal size.
- Each beam is dynamically steered toward one cell.
- No frequency reuse: all beams use the same frequency resources.
- All satellites operate on the same bandwidth
- The bandwidth is time-shared among all cells, requiring scheduling.



Notations :

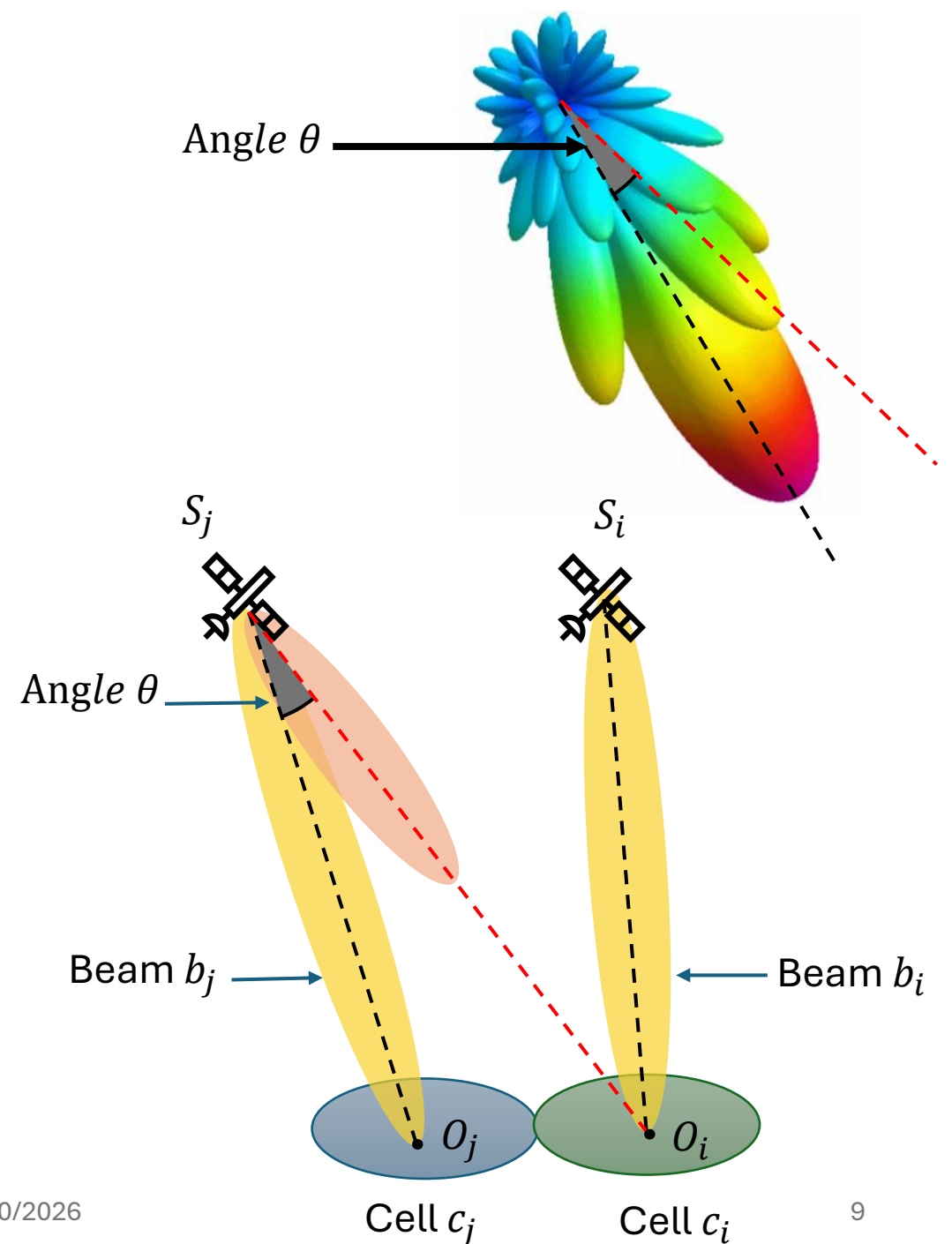
- c_i : cell i
- b_i : beam serving cell c_i
- S_k : satellite k
- p_j^{Tx} : Beam j transmission power (assumed to be equal for every beam)
- $g_{i,j}^{Tx}$: Antenna gain of beam j towards cell i
- g_i^{Rx} : the received antenna gain (assumed identical for all users)
- In 3GPP, the antenna gain for a reflector with circular aperture is given by :

$$g_\theta = \begin{cases} 1, & \theta = 0^\circ, \\ 4 \cdot \left| \frac{J_1(k_0 \cdot a \cdot \sin \theta)}{k_0 \cdot a \cdot \sin \theta} \right|^2, & 0^\circ < |\theta| \leq 90^\circ \end{cases}$$

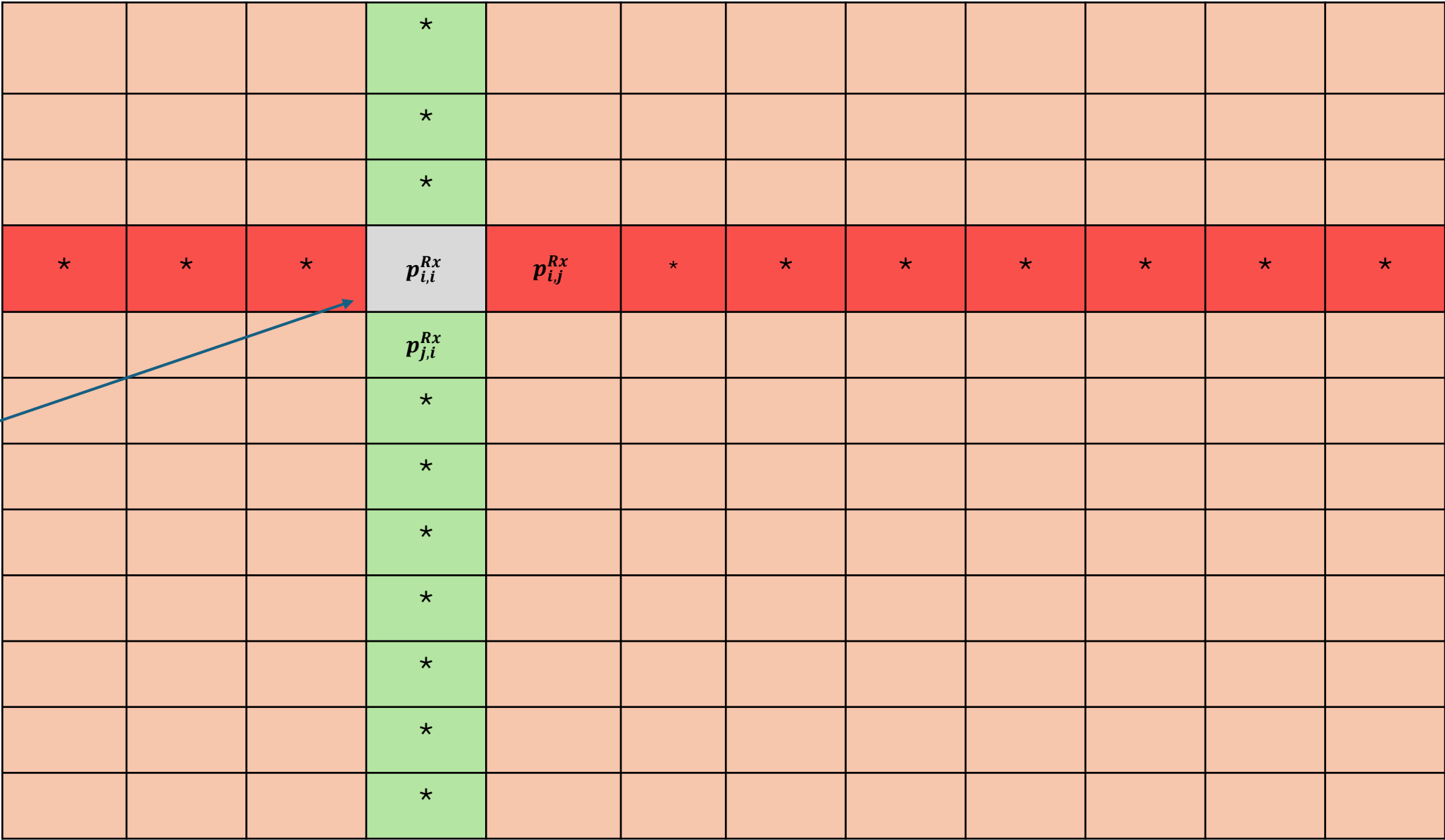
- Where :
 $J_1(\cdot)$ denotes the Bessel function of the first kind and first order
- a is the radius of antenna circular aperture
- $k_0 = \frac{2\pi f}{c}$ is the wave number (f carrier frequency and c is the speed of light)

Formula : Power received at cell i from beam j

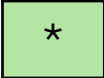
$$p_{i,j}^{Rx} = p_j^{Tx} \cdot g_{i,j}^{Tx} \cdot g_i^{Rx} \cdot \left(\frac{\lambda}{4\pi \|S_j - O_j\|} \right)^2$$



Matrix P^{Rx} :
Power
at reception



Cell of interest i
when served by
beam i

 : victims of beam i

 : aggressors to cell i

Interference Di-graph $D(P^{Rx}, \alpha)$

- Vertices : set of cells $\mathcal{C}$
- Edges : $A \subseteq \mathcal{C} \times \mathcal{C}$ is constructed by selecting the top α fractions of all cell pairs that cause the most significant interference (those with the lowest pair-wise SIR)
- Therefore, $D(P^{Rx}, \alpha)$ represents a mapping of the most significant pair-wise interference among cells.
- We can also represent the interference graph by the adjacency matrix $w(\alpha)$ such that :

$$w_{i,j}(\alpha) = \begin{cases} p_{i,j}^{Rx} & \text{if } (i,j) \in A \\ 0 & \text{otherwise} \end{cases}$$

Graph coloring optimization problem

$$\begin{aligned}
 &\underset{\mathbf{x}, \mathbf{y}}{\text{minimize}} && \sum_{k=1}^C y_k \\
 &\text{subject to} && \sum_{k=1}^C x_{i,k} = 1, \forall i \in \mathcal{C} && (1) \\
 &&& x_{i,k} \leq y_k, \forall i \in \mathcal{C}, k \leq C && (2) \\
 &&& \sum_{\substack{i \in \mathcal{C} \\ i \neq j}} \sum_{k=1}^C w_{i,j}(\alpha) \cdot x_{i,k} \cdot x_{j,k} \leq B_j, \forall j \in \mathcal{C} && (3) \\
 &&& \mathbf{x} \in \{0, 1\}^{C \times \kappa}, \mathbf{y} \in \{0, 1\}^C && (4)
 \end{aligned}$$

Variables

$x_{i,k} \in \{0,1\}$ cell i gets color k
 $y_k \in \{0,1\}$ color k is used

Parameters

$D(P^{Rx}, \alpha)$ Interference Di-graph

$\{i, j\} \in A \Rightarrow$ beam j is aggressor to cell i

$w_{ij} = p_{i,j}^{Rx}$ if $(i, j) \in A$ and 0 otherwise

$B_j \geq 0$ common conflict budget permitted at cell j

Why minimizing the number of colors ?

→ By minimizing the number of colors, we make sure that we avoid the excessive bandwidth time splitting +

We have a constraint on the interference limit

- Choice of α :
- How does α impact the performance of the algorithm ?
 - α controls the density of the di-graph :
 - $\alpha = 1 \rightarrow D(P^{Rx}, \alpha)$ is complete
 - $\alpha = 0 \rightarrow D(P^{Rx}, \alpha)$ is empty
- Scenario : 17 cells placed in a hexagonal pattern and we sweep α from 0 to 1. For each value, we construct $D(P^{Rx}, \alpha)$ and apply the graph coloring optimization.

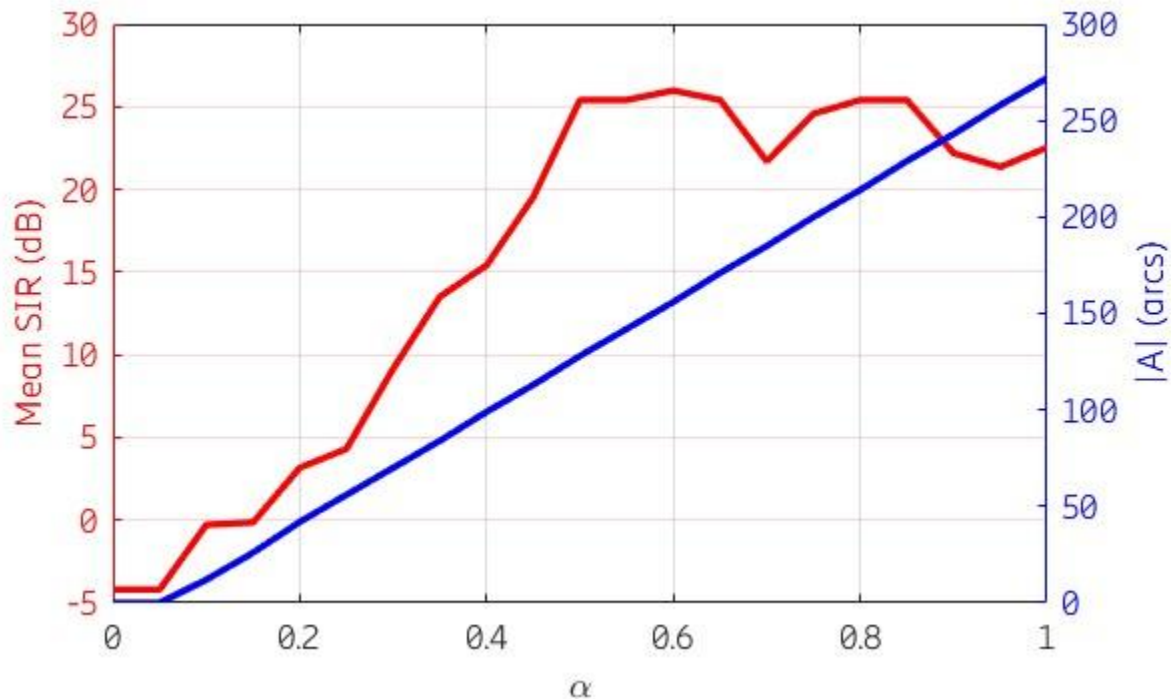


Fig 1 : Mean SIR and graph density as a function of α .

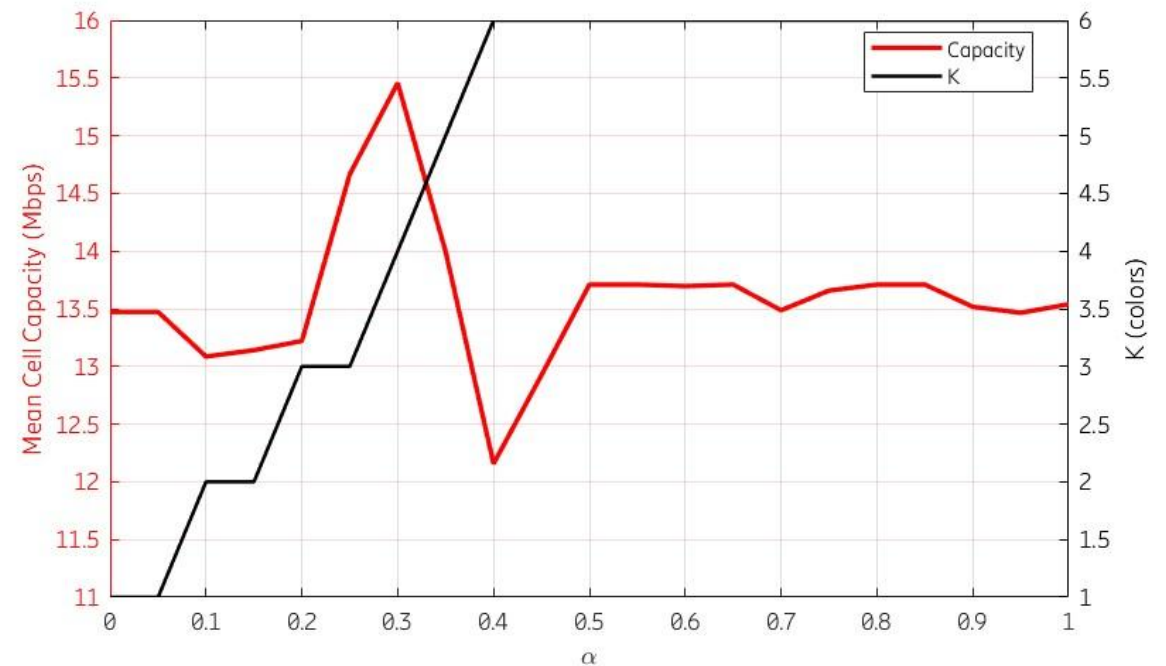
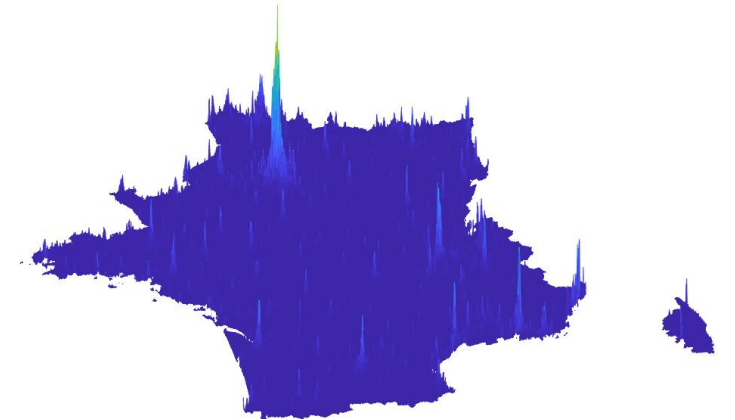
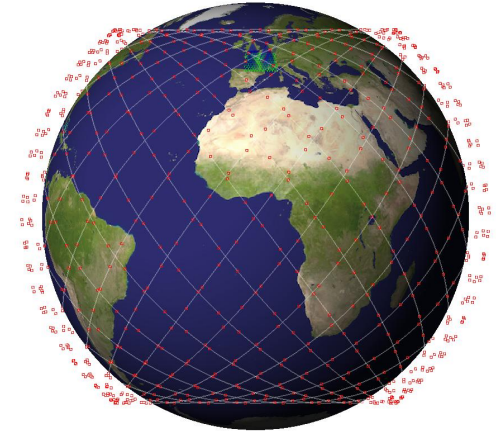


Fig 2: Mean cell capacity and K as a function of α .

Experimental setup

- 600 LEO satellites at an altitude of 440 km
- Each satellite generates beams with HPBW 4.33°
- The service link operates in L-band at a carrier frequency of 1 GHz with a bandwidth of 30 MHz
- The region of interest covers 17 cells, arranged in a hexagonal grid with an inter-site distance of 22.2 km.
- Each cell serves 0 to 50 randomly dispersed UEs.



Baseline Algorithms

- ILP-based graph-coloring solution ($\alpha = 0.25, \alpha = 0.35, \alpha = 0.70$)
Provided by Branch and Bound Algorithm
- Regular beam hopping based on geometrical separation (BH = 1, BH = 3 and BH = 4)
- Greedy search : 2-Pass Heuristic, which, given a fixed number of colors, starts from a random coloring and iteratively improves it by testing all possible color reassignments for each cell and selecting the one that minimizes the total number of constraint violations.

Algorithm 1 Greedy Coloring Algorithm

```
1: Input: Graph  $D = (\mathcal{C}, \mathcal{A})$ , colors  $K$ 
2: Initialize  $c$  randomly,  $V \leftarrow \text{VIOLATIONS}(c)$ 
3: repeat
4:    $V_{\text{old}} \leftarrow V$ 
5:   for  $i \in \mathcal{C}$  do
6:      $c_i \leftarrow \arg \min_{k \in \{1, \dots, K\}} \text{VIOLATIONS}(c_{i \leftarrow k})$ 
7:   end for
8:    $V \leftarrow \text{VIOLATIONS}(c)$ 
9: until  $V = V_{\text{old}}$ 
10: return  $(c, V)$ 
```

Metrics

- **Cell capacity** : computed analytically using the Shannon formula. For a given coloring with K colors, the capacity of cell i is defined as : $c_i(C') = \frac{\omega}{K} \cdot \log_2(1 + SINR_i(C'))$, where ω is the total bandwidth, K is the number of colors.

SINR is computed as :
$$SINR_i = \frac{1}{\frac{1}{SIR_i} + \frac{1}{SNR_i}}$$

- **Interference budget violations rate** : the number of nodes exceeding the interference budget.
- **User throughput satisfaction** : each UE is assigned a traffic demand drawn from a lognormal distribution depending on the geographical environment type (e.g., rural, urban) of its serving cell. This traffic is injected into the simulator as the requested traffic for each UE.

Numerical results

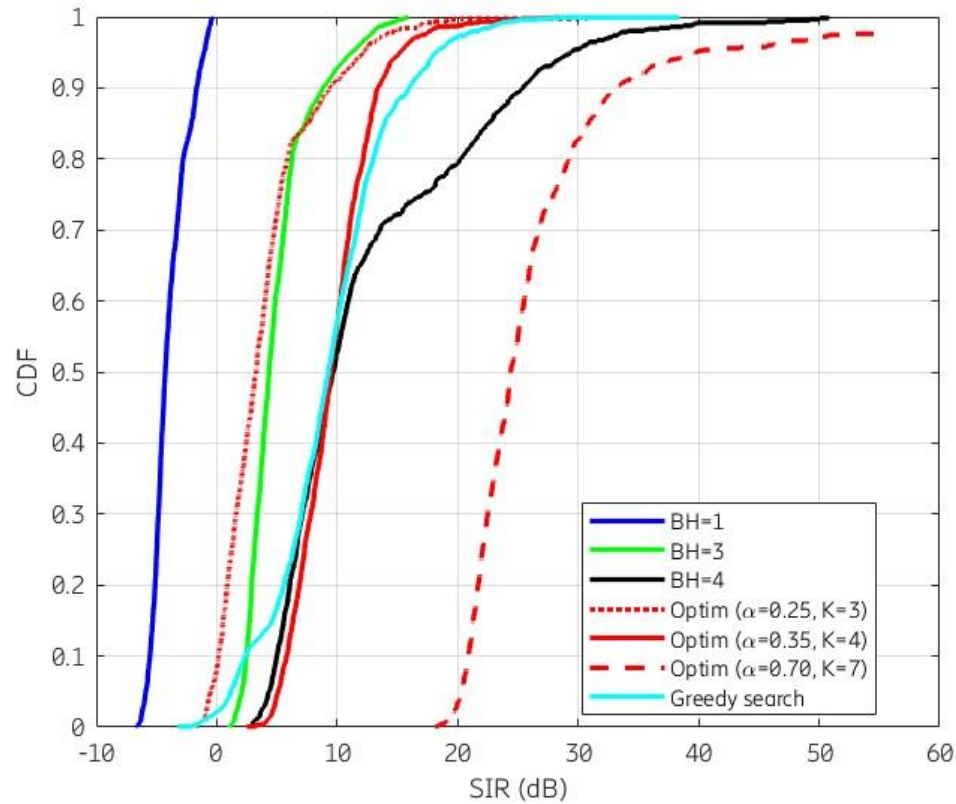


Fig 3. SIR CDFs for all methods

- Lowest SIR => BH = 1
- BH : when increasing the number of colors, SIR increases
- Optimization-based algorithm => if α is higher , the graph is denser , K is higher and SIR is higher
- Fairness : Optimization-based algorithm realize more fairness than BH because BH focuses on the best-served users (located at the center of the cells)

Numerical results

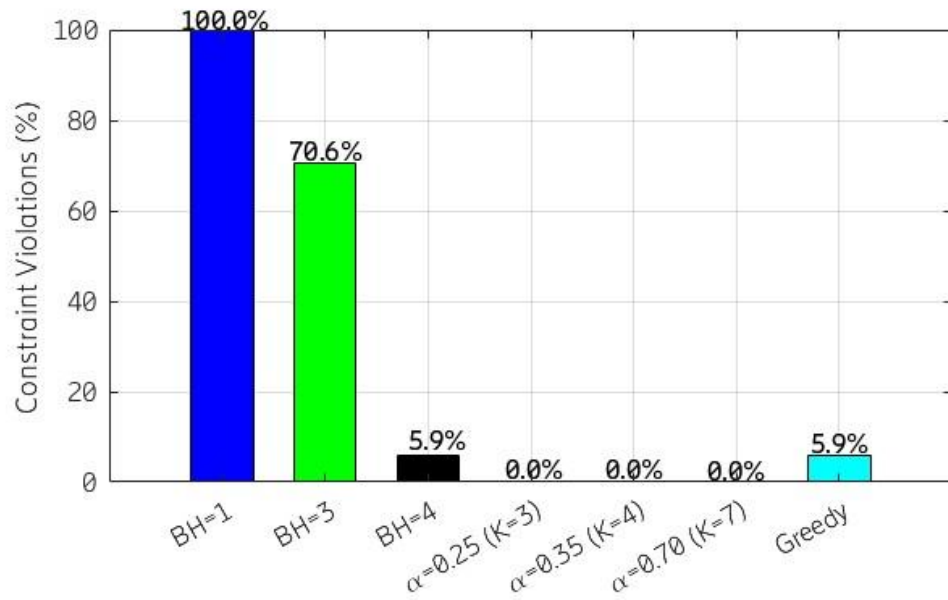


Fig 4. Constraint violations for all methods

- BH=1 exhibits 100% constraint violations.
- BH=3 less violations.
- Optimization based algorithms achieve 0% constraint violation.
- Greedy search : 5.9%

Numerical results

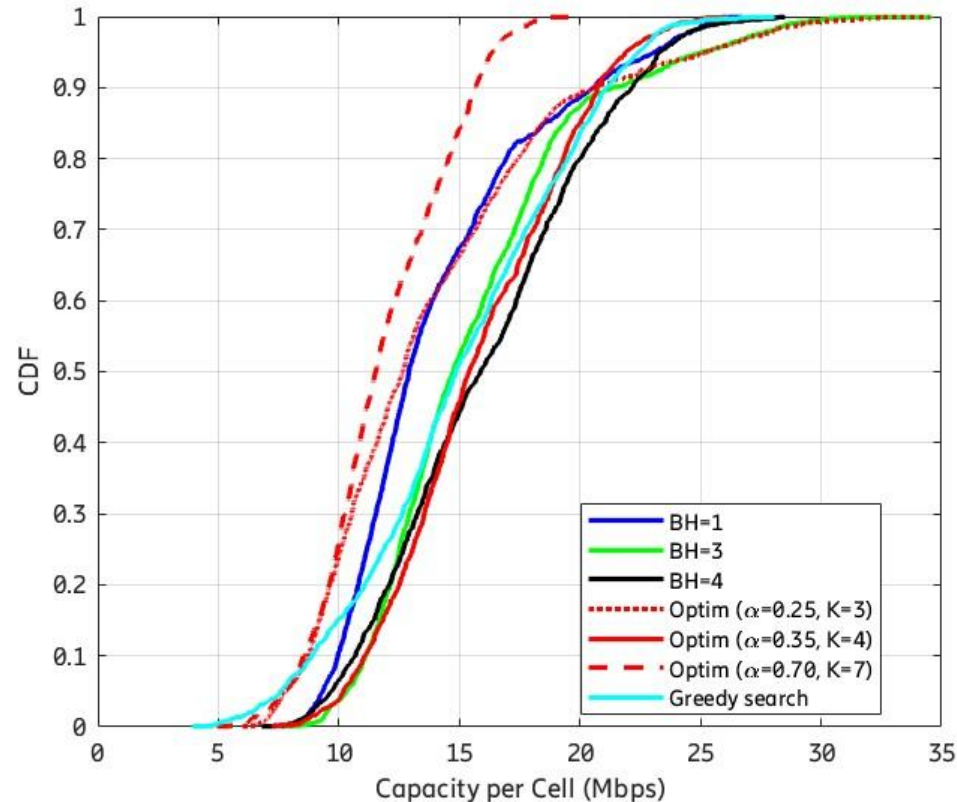


Fig 5. Capacity per cell CDFs

TABLE I
USER CAPACITY (Mbps)

Algorithm	Mean	Median	P5	P95	JFI
BH=1	0.48	0.41	0.04	1.03	0.5772
BH=3	0.51	0.42	0.04	1.25	0.5443
BH=4	0.52	0.43	0.04	1.31	0.5473
Optim ($\alpha = 0.35$, $K = 4$)	0.52	0.44	0.04	1.18	0.5623
Greedy search	0.51	0.43	0.04	1.19	0.5545

- Among the regular BH techniques, BH=1 has the lowest capacity per cell values. However, it is not the lowest :
Optim $\alpha = 0.7$ the lowest => because $K=7$ => each beam is only scheduled 1/7 slots => cell capacity is penalized.
- Optim $\alpha = 0.35$ presents comparable performance to BH = 4 with more fairness

Numerical results

TABLE II
USER TRAFFIC SATISFACTION (SERVED/OFFERED)

Algorithm	Mean	Median
BH=4	0.452	0.485
Optim K=4	0.460	0.530
BH=3	0.443	0.455

- Optim $\alpha = 0.35$ presents the best mean and median number for UE satisfaction rate.

Conclusion

- We proposed an interference-aware Beam Hopping scheduling framework for LEO satellite networks based on soft-conflict graph coloring that provides a good balance between limiting interference and avoiding excessive time-slot fragmentation.
- Unlike conventional Beam Hopping schemes that rely on predefined geometric patterns, the proposed algorithm rely on physical interference to take scheduling decisions.
- Adaptable approach to heterogeneous and irregular NTN deployments where interference relationships may be highly irregular and difficult to capture through fixed reuse patterns.

Future work

- Cell planning for NTN cells

Thank you

References

- [1] : Yaxin Wang et al. “A Flexible Resource Allocation Algorithm in Full Bandwidth Beam Hopping Satellite Systems”. In: 2019 IEEE 3rd Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC). 2019
- [2] : Zhiyuan Lin et al. “Multi-Satellite Beam Hopping Based on Load Balancing and Interference Avoidance for NGSO Satellite Communication Systems”. In: IEEE Transactions on Communications 71.1 (2023)
- [3] : Ruishuo Mu et al. “Dynamic Beam Hopping for LEO Satellites with Differentiated Traffic Demands”. In: WSCP2023. 2023, pp. 621–626. DOI: 10.1109/WCSP58612.2023.10405252
- [4] : Huimin Deng et al. “Satellites Beam Hopping Scheduling for Interference Avoidance”. In: IEEE Journal on Selected Areas in Communications 42.12 (2024),
- [5] : Zijun Liu et al. “Multi-Satellite Coordinated Beam Hopping for Interference Mitigation Under Tilted Beam Effects: A Graph-Theoretic Approach”. In: IEEE Wire less Communications Letters 15 (2026), pp. 2313–2317. DOI: 10.1109/LWC.2026.3676112.

What is the budget for each cell $B(c_i)$?

$$\frac{P(b_{c_i} \rightarrow c_i)}{\sum_{j \neq i} P(b_{c_j} \rightarrow c_i)} \geq SIR_{threshold_linear}$$

$$\frac{P(b_{c_i} \rightarrow c_i)}{SIR_{threshold_linear}} \geq \sum_{j \neq i} P(b_{c_j} \rightarrow c_i) \quad (6) \quad \sum_{i: \{i,j\} \in E} \sum_{k=1}^K w_{ij} z_{ij,k} \leq B(i) \quad \forall i \in V$$

$$B(c_i) = \frac{P(b_{c_i} \rightarrow c_i)}{SIR_{threshold_linear}} = \frac{P(b_{c_i} \rightarrow c_i)}{10^{\frac{SIR_{threshold_dB}}{10}}} = P(b_{c_i} \rightarrow c_i) 10^{\frac{-SIR_{threshold_dB}}{10}}$$