

MIT Open Access Articles

Static beam placement and frequency plan algorithms for LEO constellations

The MIT Faculty has made this article openly available. **Please share** how this access benefits you. Your story matters.

Citation: Pachler de la Osa, Nils, Guerster, Markus, Portillo Barrios, Inigo, Crawley, Edward and Cameron, Bruce. 2020. "Static beam placement and frequency plan algorithms for LEO constellations." International Journal of Satellite Communications and Networking, 39 (1).

As Published: 10.1002/sat.1345

Publisher: Wiley

Persistent URL: <https://hdl.handle.net/1721.1/137279>

Version: Author's final manuscript: final author's manuscript post peer review, without publisher's formatting or copy editing

Terms of use: Creative Commons Attribution-Noncommercial-Share Alike



SPECIAL ISSUE PAPER

Static beam placement and frequency plan algorithms for LEO constellations

Nils Pachler de la Osa^{ID} | Markus Guerster | Inigo del Portillo Barrios |
Edward Crawley | Bruce Cameron^{ID}

Aeronautics and Astronautics, Massachusetts
Institute of Technology, Cambridge,
Massachusetts, USA

Correspondence

Nils Pachler de la Osa, Aeronautics and
Astronautics, Massachusetts Institute of
Technology, Cambridge, Massachusetts, USA.
Email: pachler@mit.edu

Summary

LEO satellites are growing rapidly in number, but also in configurability of the payload, which presents new challenges of how to allocate the satellite's resources. We present an algorithm to solve two subproblems of the resource allocation problem for multi-beam satellites in large LEO constellations. Our approach produces a static beam placement and a static frequency plan, taking into account the user demands and interference conditions, respectively. We provide an easy to implement and fast approximation for the beam placement problem, which is converted into an *Edge Clique Cover* problem using graph theory. The frequency assignment problem is translated into a constraint satisfaction problem, which is later solved with a first-fit heuristic. We apply our algorithm to a scenario based on SpaceX's Starlink constellation, and when compared to two different baselines, we show that our method requires 40% fewer beams while being able to provide service to 2% more users.

KEYWORDS

beam placement, frequency assignment, LEO constellations, satellite resource allocation, Starlink

1 | INTRODUCTION

Low Earth orbit (LEO) constellations for satellite communications are regaining popularity since their decline in the 1990s.¹ In the last years, up to 11 companies have filed their FCC proposals for NGSO-type constellations. Driven by a higher throughput than their predecessors,² these fleets are expected to disrupt the market within the next years. The increased performance achieved by the new payloads is a combination of several factors including, but not limited to, digital payloads, advanced modulation schemes, and multi-beam antennas. Furthermore, these new constellations are designed with higher flexibility, allowing for the satellites to dynamically adapt their configuration and optimize the throughput delivered.

However, this flexibility comes at the expense of increased complexity. Manual allocation of resources, extended during the early years of the industry, becomes no longer feasible with the large set of variables and higher dimensionality introduced in modern constellations. Instead, automated dynamic resource allocation methods will be necessary to fully exploit the capabilities of these novel systems. The objective of this paper is to address these needs by solving the beam placement and frequency assignment problems.

Modern communication payloads have four categories of resources that can be individually configured for each beam³: pointing direction, transmission frequency (bandwidth and central frequency), shape, and transmitted power. Each of these resources is subject to its own set of constraints. For example, there is a maximum number of beams that can be generated, and their shape (normally circular or elliptical) can only be adjusted within certain limits. The satellite has a finite pool of frequency spectrum available and is limited in power. Moreover, factors such as user demand or interfering effects among beams need to be considered in the resource allocation process.

In order to simplify the resource allocation (RA) problem, we decouple the allocation of different resources and treat them separately. In this paper, we focus on the beam placement and frequency assignment subproblems and present an approximation to both of them—with particular focus on LEO constellations.

1.1 | Literature review

From the four resources introduced above, recent research has focused on power allocation,⁴⁻⁸ while the other satellite resources such as the frequency assignment and beam placement have been less studied. The latter two problems have been shown to be NP hard,^{9,10} which is why brute force algorithms are unfeasible for medium- to large-scale scenarios (ie, >850 beams, as we show later).

For the beam placement, Camino et al^{11,12} approached the problem with mixed-integer linear and linearization techniques and provided solutions for up to 300 user terminals. Although optimal, finding a solution for larger scenarios is computationally infeasible. On the other hand, Kyrgiazos et al¹³ divided the Earth's surface into a polygonal layout and adjusted the beam shape and pointing to achieve a load-balanced distribution between beams. Their approach uses a self-organizing feature map combined with the K-means algorithm to find the placement of the beams and reaches a solution for low number of beams (up to 200). This technique, however, requires prior knowledge of the number of beams and does not guarantee the constraint satisfaction of the problem.

As we illustrate, the beam placement problem can be transformed into an *Edge Clique Cover* problem in graph theory, which aims to cover a set of points with a set of graphs. This problem has been extensively studied in the literature outside of the field.¹⁴⁻¹⁷ However, the aim of these articles is to find the optimal solution to the problem, which is not feasible for high-dimensional problems as the one studied in this work.

For the frequency assignment problem, some authors have approached the solution using mathematical programming (binary search, Park et al^{18,19}; convex optimization, Wang et al²⁰). Using as a reference the water filling method,²¹ these papers provide alternative solutions that lead to fairer or higher capacity distributions. The solutions presented show improvements for low number of beams but are computationally unfeasible for larger data sets. Other authors, using the interference power as a figure of merit, have developed artificial intelligence algorithms to solve the frequency assignment problem (Deep Reinforcement Learning, Hu et al²² and Liu et al²³; Gradual Neural Network, Funabiki and Nishikawa²⁴ and population-based algorithms, Salcedo-Sanz et al^{25,26}). These approaches are based on the optimization of co-channel interference between two nearby satellites. The frequency plan for one satellite is considered fixed, and the algorithm changes the frequency allocation for the second one. While the solutions proposed show improvements with respect to previous methods, they do not consider more than two satellites, which does not make them suitable for LEO constellations.

No work has been found regarding the beam placement and frequency assignment problems in communications satellites applied to LEO constellations. The transfer of approaches developed for GEO is often not trivial, since LEO constellations introduce time dependency, handovers, and a higher number of variables to the resource allocation problem.

1.2 | The resource allocation problem

The resource allocation (RA) problem is to optimally assign one or more resources from a limited pool to a set of resource-requiring entities. In satellite communications, the *satellite's resources* have to be distributed among *users* to satisfy their *demand*. In particular, the RA problem within the satellite communications context consists of setting the direction, frequency, shape, and power for each beam. Due to the large number of beams used in modern constellations, this represents a high-dimensional problem, which is challenging to approach with conventional optimization techniques.

In non-geostationary orbit (NGSO) constellations, the network topology is constantly changing, which adds an additional layer of complexity to the problem, as an explicit time dependency component needs to be considered. In addition to the user-demand variation found in GEO, there are two main sources of variation over time: the changes in the satellite's orbital positions and handover between satellites.

However, current technology makes continuously changing all the resources difficult. While a change in power is almost instantaneous, a change in the beam's shape or pointing direction incurs a higher time penalty, making them undesirable. Similarly, power changes are preferred over frequency ones, as user terminals need to reacquire the signal, lock in, and synchronize.

New satellite constellations require many more satellites than their MEO or GEO counterparts to provide global coverage. For the RA problem, however, the large number of LEO satellites adds complexity. For MEO and GEO constellations, user terminals tend to have one or two satellites within field of view (FOV), whereas in LEO constellations, there are dozens of satellites in the FOV. This diversity of satellites introduces a new variable: the mapping of users and beams to satellites.

In addition, the variables to optimize (pointing direction, frequency, shape, power and *satellite* for each beam) present several differences in the LEO context compared to MEO and GEO orbits. The low altitude for LEO orbits results in smaller beam footprints, and hence fewer user terminals can be covered by one beam. This directly translates to more beams, which increases the dimensionality of the RA problem.

1.3 | Overview of the proposed approach

All these factors pose challenges to the RA problem, as a large number of constraints need to be considered by the resource allocation model. In order to simplify the problem, we make several assumptions: First, we assume a static beam placement and frequency assignment, ie, once assigned, the location and frequency of a beam are fixed. Second, the beam placement and frequency assignment problems are independent and can be solved individually. Third, we only assume downlink traffic for the user link. Finally, we do not consider the beam shaping problem; instead, we assumed that all beam shapes are conical with a constant half-power-beam-bandwidth (δ).

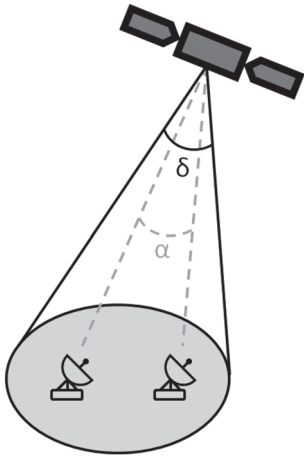


FIGURE 1 Representation of the angular separation between users α and beam aperture angle δ . In this case, the two users could be in the same beam

The approach proposed in this paper is based on a three-stage implementation: First, the beam placement problem is solved based on the locations of the users. The output of this stage is a set of latitude and longitude coordinates of the beams. Second, the frequency assignment is generated based on the beam locations previously computed and the peak demand of the users. The output of this stage is a downlink frequency plan, where each beam is assigned a bandwidth and a central frequency. Finally, power is dynamically allocated to each beam, producing a full resource allocation solution.

Since power allocation has been extensively studied in the literature, this paper focuses on the first two stages defined above. The remainder of this paper is structured as follows. Section 2 covers the first stage: the beam placement problem. Section 3 formulates and solves the frequency assignment problem. Section 4 presents the results of our algorithms for an scenario based on SpaceX's Starlink constellation. Finally, Section 5 summarizes the conclusions and outlines areas of future research.

2 | BEAM PLACEMENT

This section focuses on the first stage: beam placement. Section 2.1 formulates the problem and converts it into an *Edge Clique Cover* representation. Then, in Section 2.2, we describe our approach for solving the problem, which is based on a heuristic approximation.

2.1 | Problem definition

In our beam placement problem, the objective is to cover all users minimizing the number of beams. The altitude of the satellite together with the half-cone angle ($\frac{\delta}{2}$) of the beam determines the footprint of the beam. If users are close enough, they can be covered by the same beam. Formally, we say that given a satellite's position and altitude, two users are separated by an angle α (see Figure 1).

Two users can only be in the same beam if $\alpha \leq \delta$. However, α changes with the position of the satellite, and therefore, a time dependency is introduced: Two users can only be assigned to the same beam if $\alpha \leq \delta$ for all possible positions of the satellites. The maximum value of α determines whether the inequality holds or not, leading to $\alpha_{max} \leq \delta$. This worst case scenario angle has to be computed for each pair of users and can be expressed in the form of an adjacency matrix. Equation (2.1) gives the mathematical representation of the matrix in the case of four users. From the definition, this matrix is symmetrical and the diagonal is 0. Given δ , this matrix can be transformed into a Boolean adjacency matrix, where a value of 1 indicates that two users can be in the same beam, and a value of 0 that they cannot.

$$\begin{array}{c} \text{Users} \\ \begin{bmatrix} 0 & \alpha_{12} & \alpha_{13} & \alpha_{14} \\ \alpha_{21} & 0 & \alpha_{23} & \alpha_{24} \\ \alpha_{31} & \alpha_{32} & 0 & \alpha_{34} \\ \alpha_{41} & \alpha_{42} & \alpha_{43} & 0 \end{bmatrix} \end{array} \quad (1)$$

This adjacency matrix defines an undirected graph, where each user becomes a node, and an edge between two nodes represents that those two users can be covered by the same beam. Therefore, for a set of users to be covered by a single beam, they must all be pairwise connected (the angular separation between any pair of users is smaller than δ), forming a *clique*.

The graph shown in Figure 2A introduces an example that we will use throughout the paper. We show a scenario with 20 users. Figure 2B corresponds to the adjacency matrix of the first six users in the example. Some of the possible beams cover these users: 1-2, 2-3-4, 2-3, 2-4, 3-4, 1, 2, 3, 4, etc. Out of these, some beams fully overlap with other beams (for example, 2-3-4 overlaps with 2-3 and 2-4). In these cases, we only consider the maximum cliques, ie, the beam with the most users, here 2-3-4 using the previous example. This does not compromise the optimal

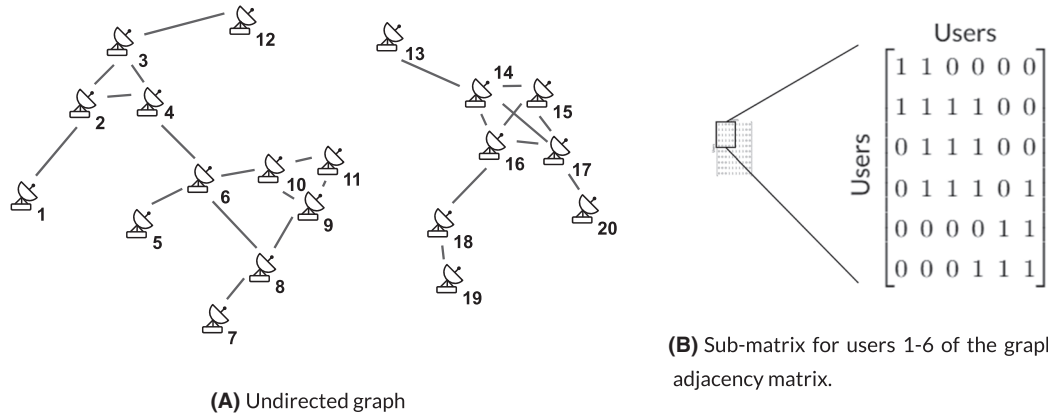


FIGURE 2 User terminal arrangement and undirected graph for an example case with 20 user terminals. Two user terminals can be covered by the same beam if they are connected by an edge. There are 15 maximum cliques in the graph

solution, as the discarded cliques are a subset of the ones considered. Therefore, the beam placement problem can be stated as finding the shortest list of maximum cliques that covers all the users, effectively transforming the beam placement problem into the *edge clique cover* problem.

2.2 | Proposed algorithm for the beam placement problem

Finding the optimal solution of the edge clique cover problem is known to be NP hard.²⁷ However, we focus on finding a *good enough* solution that is fast to compute. Hence, we apply an incremental algorithm based on heuristics as described in the following lines and illustrated in Algorithm 1.

As a first step, we sort the maximum cliques by size in descending order. The reason is that the size of the cliques is directly proportional to the users covered. Therefore, we prioritize those beams with a high number of users covered (higher order cliques). For cliques with the same size, we randomize the order, which makes our algorithm non-deterministic.

After sorting the cliques, we start our incremental approach. We initialize an empty list that is used for saving the cliques of the current solution. Next, we iterate through the cliques and check for collision. We define a collision between two cliques whenever there are shared vertices in both cliques. The degree of the collision is determined by the number of shared vertices. For each clique, we apply the following logic: If the clique considered does not collide with any other clique in the current solution, add it to the list. Once the algorithm finishes iterating through all cliques, a non-collision intermediate solution is found. However, this solution may not be complete, as not all the terminals may be covered. If that is the case, we do additional passes allowing cliques with a higher degree of collision. When we add a colliding clique to the solution, we reduce the size of the clique to avoid having user terminals covered by several cliques. We start with an allowed degree of collision of zero and terminate when a complete solution is found.

The approach presented achieves different quality results depending on the clique ordering. To avoid local optima, we run the algorithm several times, randomizing each time the order of the same-size cliques. The number of times the algorithm is run will be denoted as *number of runs*.

Algorithm 1 Solution construction

Input: cliques Input: n_users Output: solution 1: Sort(cliques) 2: $S = []$ 3: $c = 0$ 4: while $ S < n_users$ do 5: for clique in cliques do 6: if $ clique \cup (S) > S $ and $ clique \cap (S) \leq c$ then 7: $S = S \cup [clique - (S)]$ 8: end if 9: end for 10: $c = c + 1$ 11: end while	// List of maximal cliques of the user's graph // Number of user terminals // List of maximal cliques that form the solution // Sort the cliques by size in descending order // Initialize empty solution (S) // Initialize collision order (c) // Check solution completeness // If clique is not in solution and the collision order is lower than the threshold // Add maximum non-colliding clique to the solution // Increase collision order
---	---

We applied this algorithm to the prior example from Figure 2). The solution found by our algorithm is shown in Figure 3. In this case, the solution is optimal, ie, there is no way to cover the set of 20 users with less than ten beams. Depending on the ordering in the cliques, however, there exist non-optimal possible solutions. This can be seen if the clique 6-8 is ordered before 5-6 and 7-8. For this case, the algorithm would lead to a suboptimal solution with 11 beams, since users {5, 6, 7, 8} would be covered with three beams instead of the current two. The number of runs is used to decrease the probability of these suboptimal orders.

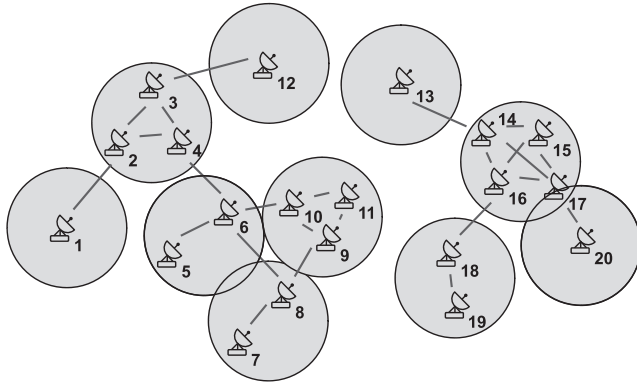


FIGURE 3 Beam placement solution found by the algorithm. 10 out of the 15 maximum cliques are selected, and thus the 20 users can be covered with 10 beams. Even though the solution displayed is optimal (has minimum number of beams), the algorithm might produce other suboptimal solutions due to randomization when sorting cliques of the same size

3 | FREQUENCY ASSIGNMENT

After describing the solution to the beam placement problem, we describe in this section the second stage of the approach: the assignment of frequency resources to beams, known in the literature as the *frequency assignment* or *frequency planning* problem. The problem is presented in Section 3.1 and then translated into a constraint satisfaction problem in Section 3.2. An algorithm to solve the frequency assignment problem using a heuristic approach is introduced in Section 3.4.

3.1 | Problem definition

The objective of the frequency assignment problem is to assign a central frequency and bandwidth to the maximum number of beams, while satisfying the interference and frequency reuse constraints. Furthermore, we want to use as much of the available bandwidth on the satellite as possible, ie, more bandwidth per beam and lower power consumption for the same data rate. In current systems, the spectrum is subdivided into frequency channels. The same frequencies (ie, overlapping channels) can be reused multiple times as long as they do not interfere. This is called the reuse factor N_r , which depends on the payload's design. Moreover, two beams can overlap in frequency and space without interfering as long as they use orthogonal polarization, which effectively doubles the available spectrum.

$$N_{ch} = N_p N_r \frac{\Delta F}{C_B}. \quad (2)$$

The total number of frequency channels available in a system can be computed by Equation (2), where ΔF is the bandwidth available, C_B is the channel bandwidth, N_r is the frequency reuse factor, and N_p accounts for the number of polarizations used (1 or 2).

3.2 | Frequency assignment constraints

This section introduces the two frequency constraints that we use to formulate the constraint satisfaction problem. The frequency interference constraint is explained in Section 3.2.1, while the frequency reuse constraint is explained in Section 3.2.2.

3.2.1 | Frequency interference constraint

Two beams only interfere if they have the same frequency and polarization and they point to *nearby* locations. Although the interference depends on the antenna's patterns, we define a threshold angular separation between two beams for which we say interference is negligible. Two beams, then, may have the same frequency only if their angular separation is higher than this threshold. For the frequency assignment, the angular separation between two beams will be denoted by β .

In a similar way as in the beam placement problem, this angle is computed for the worst case scenario, which is here the *minimum* value of β . The inequality that must hold is

$$\beta_{min} \leq \epsilon, \quad (3)$$

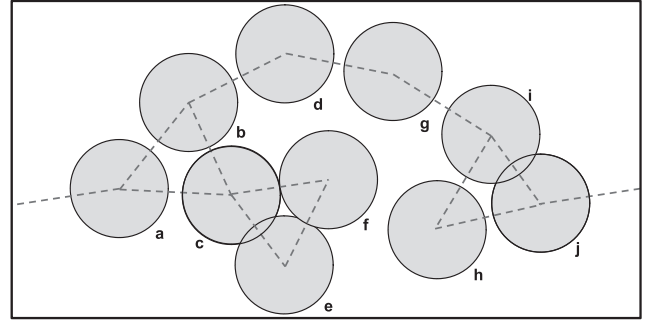
where ϵ represents the threshold angular separation between beams. Once β_{min} is determined for all beams, an adjacency matrix is constructed, setting a value of 1 when the inequality holds and 0 otherwise.

Continuing with our running example, the frequency interference constraint between beams can be computed, which results in the graph shown in Figure 4. This example assumes that the graph shown is repeated in space, ie, the arrangement of the beams is repeated left and right of the figure, so that j has a frequency restriction with a beam identical to a and idem for a .

3.2.2 | Frequency reuse constraint

The frequency assignment solution has to ensure that the reuse factor respects the reuse factor limit at any time. For this reason, we define a frequency reuse constraint (identified with the subindex r) for any two beams that are *at some point* managed by the same satellite. This information can be represented as a look-up table, where the connections state the existence of this frequency reuse constraint for a given pair of beams.

FIGURE 4 Graph containing the frequency interference constraints for the beams in the example. Two beams connected by an edge would interfere if assigned overlapping frequencies and the same polarization. The beams represented form a repeating patten than extend to the left and right of the figure



All the constraints are satisfied if, for each beam b of frequency f , there exist at most $N_r - 1$ other beams that use the same frequency and have a frequency reuse constraint with b :

$$N_{r,f} < N_r, \quad (4)$$

where $N_{r,f}$ identifies the number of beams that have frequency reuse constraint with b and have the same frequency f . As the frequency is divided in channels, this condition must be true for all the frequency channels in f .

3.3 | Number of frequency channels computation

In addition to the interference and frequency reuse constraints, the bandwidth assigned to each beam b (N_b) is required to solve the frequency assignment problem. Our algorithm starts with an initial value and reduces it, if necessary, to maximize the number of beams allocated. N_b is an upper bound limited when the satellite is under maximum load, which means that, for every time-step, the load for every satellite needs to be known and only the worst N_b should be considered. However, this cannot be known without solving the frequency assignment problem, which is the objective of this formulation. To avoid this circular dependency, we use an optimistic estimation as N_b initial value.

This optimist approach is based on the *ideal case*, which is represented by a balanced distribution of demand (ie, the load of the satellite is constant over time). In other words, this means that when a satellite loses the line of sight to one beam, another beam with identical demand becomes visible. A beam will have frequency reuse constraints with a set of beams whose total demand is D . Half of these beams will be present in the satellite when the beam of interest first connects to the satellite, while the other half will be present when it is about to disconnect. Therefore, at any point in time, only $\frac{D}{2}$ will be active, and the maximum value for N_b can be computed as

$$N_b = \rho N_{ch} \frac{D_b}{D_b + \frac{D_r}{2}}, \quad (5)$$

$$D_r = \sum_{b' \in r, b' \neq b} D_{b'}, \quad (6)$$

where ρ represents an allocation factor (between 0 and 1) that is initialized to 1 and can be adjusted when the system is bandwidth constrained, N_{ch} is the total number of the satellite's channels computed in Equation (2), D_b represents the peak demand of the beam b , and D_r is the demand sum of all the beams the beam b has a frequency reuse constraint with (not including itself).

Since the demand distribution is not ideal, the number of frequency channels assigned will be lower than the result of Equation (5).

3.3.1 | The satellite-group as a characteristic of LEO constellations

Up to this point, we have assumed that each terminal is assigned to one satellite at any given point time, and this information is known to the algorithm. While this assumption is valid for MEO and GEO constellations, it is not the case for large LEO constellations, as any user terminal has many satellites in FOV. Given the large number of satellites in FOV, the frequency reuse constraint becomes overly conservative, as the formulation protects the channel in all the satellites in the FOV, rather than just the necessary ones.

To avoid this problem, we divide the satellites into groups so that the satellites in each group provide global coverage of the surface of the Earth.

This introduces a new variable to the frequency assignment algorithm: the satellite-group. Each beam has to be assigned to a satellite-group and will only connect to the satellites of that group at all time. A beam can only be assigned to a certain group if *there is, at all times, at least one satellite of that group within its FOV*.

Given that a beam can be assigned to satellites in different groups, Equation (5) underestimates the number of channels available and needs to be multiplied by N_g to account for the different satellite-groups, resulting in

$$N_{cc} = N_{ch} N_g, \quad (7)$$

$$N_b = \rho N_{cc} \frac{D_b}{D_b + \frac{D_r}{2}}, \quad (8)$$

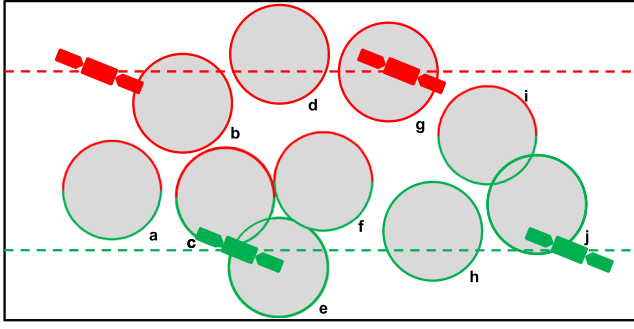


FIGURE 5 Satellite-group scheme for the beams in the running example. We assume two satellite-groups {red and green}. The beam's edge color represents the satellite-group that beam can be assigned to [Colour figure can be viewed at wileyonlinelibrary.com]

$$N_g = |\cup \mathcal{G}_b| \quad \forall \text{binr}, \quad (9)$$

where \mathcal{G}_b are the satellite-groups of beam b and N_g is the union of the satellite-groups in any beam that shares a frequency reuse constraint with the beam of interest as shown in Equation (8). This is due to the a priori unknown beam-to-satellite-group association.

For our running example, we have assumed two satellite-groups (red and green) as shown in Figure 5. The color(s) of the circumference indicates the satellite-groups for each beam, eg, $\mathcal{G}_d = [\text{red}]$, while $\mathcal{G}_g = [\text{red, green}]$.

3.4 | Proposed algorithm for the frequency assignment problem

Our approach is based on a heuristic that works the following way: Take a non-assigned beam, look for an unassigned channel that respects the constraints and assign it to the beam. If there are not enough unassigned channels that respect the constraints, the beam is marked as not assignable. We sort the beams based on the number of constraints (descending order) and iterate over them applying the logic described above. The cost of this assignment is low, as we only traverse each beam once. Furthermore, we consider three algorithms to guide the search for the next unassigned channel:

- **First-fit:** Assigns the first free channel from the lowest utilization satellite-group available, reusing frequency if possible (to avoid adding unnecessary frequency interference constraints). For equally empty satellite-groups, the order is randomized.
- **Recoloring:** Initializes all the beams with the same channel and satellite-group. Then, uses first-fit to reassign the channel and satellite-group of colliding beams.
- **Random assignment:** Tries to assign a random channel and satellite-group to a beam. If these results on a collision, tries again (up to 100 trials).

At the end of the loop, we compute how many beams have not been assigned. If all of them are assigned, the algorithm terminates. Otherwise, we use binary search to find the value of the *allocation factor* (ρ) that maximizes the number of beams allocated.

Algorithm 2 AssignFrequency

Input: beams	// List of ordered beams to assign
Input: ρ	// Allocation factor
Output: NAB	// Number of not assigned beams
1: NAB = 0	// Initialize number of not assigned beams
2: if Recoloring then	// Using recoloring algorithm
3: for beam in beams do	
4: AssignFirst(beam)	// Assign the first free channel and satellite-group
5: end for	
6: end if	
7: for beam in beams do	
8: $\mathcal{C} = []$	// Initialize empty constraint set
9: for constraint in beam do	
10: $\mathcal{C} = \mathcal{C} \cup \text{constraint}$	// Add constraint to set
11: end for	
12: AssignChannel(beam, \mathcal{C} , $N_b * \rho$)	// Get channel assignment (first-fit or random) with \mathcal{C} and new N_b
13: if not Assigned(beam) then	// Add it to the non-assigned beams if could not be assigned
14: NAB = NAB + 1	
15: end if	
16: end for	

Algorithm 2 describes the frequency assignment algorithm, as explained previously in this section. For the first-fit, the channel-assigning function assigns $N_b * \rho$ channels to the beam of interest from the emptiest satellite-group. This function prioritizes frequency reuse over using different frequencies to avoid getting stuck in avoidable frequency interference constraints. For the random assignment, it tries several channels and satellite-groups (up to 100, randomly chosen) to allocate to the beam.

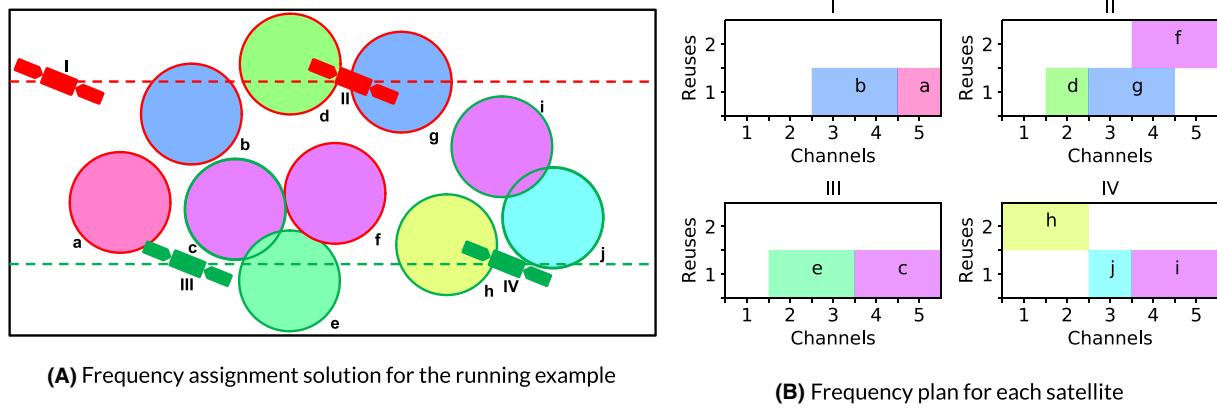


FIGURE 6 Frequency assignment solution for the beams in the running example. The beam's fill color represents its frequency band, whereas the edge color represents its satellite-group. Note that even though beams *c* and *f* have overlapping frequencies and are in close proximity, they do not interfere because they are assigned to different satellite-groups (green and red, respectively), and therefore, user terminals point in different directions [Colour figure can be viewed at wileyonlinelibrary.com]

In our running example, we assume that each user has a maximum demand of 1 unit, creating an unbalance in the demand of the beams. Also, we assume that each satellite has five frequency channels, a frequency reuse factor of 2, and a single polarization, leading to 10 channels per satellite. The solution computed by the algorithm is shown in Figure 6. It can be seen that the algorithm maintains “fairness” among satellite-groups, as all of them have similar demands. The solution showed only represents one static position of the constellation. The beam-to-satellite assignments change over time.

An example for the frequency constraint is shown by *a* and *b*. They are assigned to different frequencies in satellite I. Another example is *b* and *d*, where each beam is assigned to a different satellite. For the frequency reuse constraints, *e* and *h* have a constraint, which is why *h* is assigned the second reuse. It would otherwise collide with *e* when *h* is handed over from satellite IV to III. On the other hand, *c* and *i* are separated far enough and do not have this limitation and can benefit from the same reuse. Finally, due to their demand, the number of channels assigned to each beam is different. The initialization aims for a fair distribution among beams, giving one channel to beams with lower demand and two to those with higher demand.

4 | RESULTS

We described in the previous sections how our algorithms for the beam placement and the frequency plan work. In this section, we present realistic full-size simulation results for an scenario based on SpaceX's Starlink constellation.²⁸ Since to the best of our knowledge there is no comparable work, we implemented other, more simplistic solutions. We compare our proposed algorithms in terms of quality of the solution and computational cost with these.

Starlink constellation comprises 4409 satellites distributed across five different sets of orbits. The core of the constellation is composed of 1584 satellites evenly distributed in 72 orbital planes at 550 km with an inclination of 53°. The other 2825 satellites (part of a second deployment) are distributed as follows: 1600 satellites in 32 planes at 1110 km (inclination 53.8°); 400 satellites in eight orbital planes at 1130 km (inclination 74°); 375 satellites in five planes with 75 satellites each at 1275 km (inclination 81°); and 450 satellites in six orbital planes with 75 satellites each at 1325 km (inclination 70°). Each satellite carries a digital payload based on a phased-array that can deliver up to 23 Gbps of throughput.² The phase array uses eight downlink-channels, and we assume that a total of 32 beams can be generated (frequency reuse factor of 4). Each beam has a full-cone aperture of 4.6°, while the fields of regard of the satellites are 25° for the 550-km satellites and 40° for the rest of the satellites.

For the users, we use a traffic model provided by SES S.A., which contains the location of 18 712 projected, static user terminals distributed over the Earth. For each user, the maximum demand, longitude, and latitude are given. Users are located in the $\pm 50^\circ$ latitude band.

4.1 | Beam placement

We evaluate the performance of the beam placement algorithms by comparing it with a brute force approach, which gives the optimal solution. In addition, we implemented a baseline algorithm, which covers the Earth with a grid of circles and only maintains those with a user present. The surface of the circles is the projection of the beam on the Earth for the most constraining case (ie, the smallest beam), which for the Starlink constellation is a 4.6° cone aperture angle at 550-km altitude. To assess the scalability of our algorithms, we take subsets of the user terminals. We run the algorithm on a single core of a standard laptop.

Figure 7 shows the cost and performance of the brute force approach, the grid baseline, and the algorithm presented in this work for NR of 1, 10, and 100. The figures do not include the adjacency matrix computation, as it is invariant for a fixed user-user pair. On the other hand, the clique computation is included, as it changes depending on the number of users selected. As can be seen, the brute force algorithm is computationally

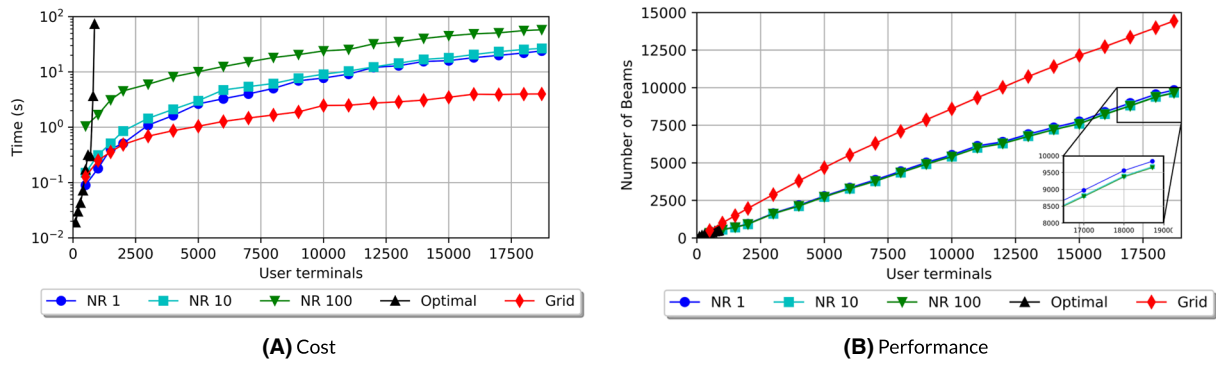


FIGURE 7 Cost and performance for different beam placement algorithms. NR stands for the number of runs in our beam placement algorithm. The zoom-in shows only a 2% reduction in the number of beams between 1NR and 10NR [Colour figure can be viewed at [wileyonlinelibrary.com](https://onlinelibrary.wiley.com)]

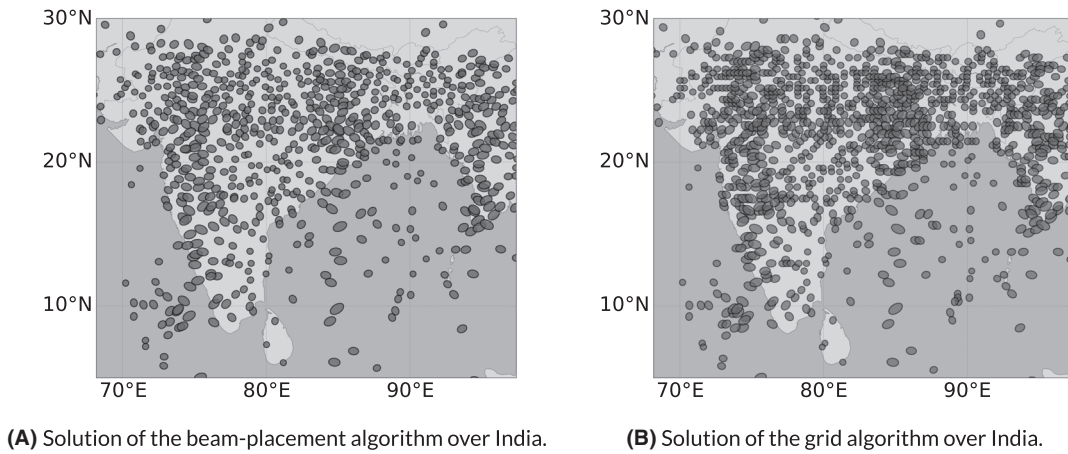


FIGURE 8 Solution of the beam placement and grid algorithms over India. 737 and 1405 beams, respectively, are required to cover 1405 user terminals. As the mapping to satellite is not yet known, each beam has been mapped to the nearest satellite in the lower orbit (550km)

unfeasible for more than 850 user terminals, as the computation time scales factorially with the size of the clique. Our algorithm has a cost of $\mathcal{O}(mn)$, where m is the size of the larger clique and n is the number of terminals. The simple grid approach has a linear computation time.

In terms of performance, our algorithm produces the optimal solution for scenarios with low number of terminals (for number of runs of 10 and 100). We cannot assess optimality for scenarios with a higher number of terminals, as the brute force algorithm becomes computationally intractable. However, for these scenarios, the results show a decrease of 40% in the number of beams with respect to the grid baseline.

Furthermore, the benefit on rerunning our algorithm multiple times can be assessed. We achieve a reduction in the number of beams of around 2% between NR1 and NR10, while requiring 10% more time. However, there is diminishing returning when going from NR10 to NR100, since the algorithm only achieves a 0.4% reduction in the number of beams while using 117% more time.

Figure 8 shows the results for the beam coverage over India for our algorithm and the grid baseline. As can be seen, the algorithm presented in this work tries to optimize the coverage of each beam and minimize overlapping, being able to cover the region (1488 users) with 791 beams. On the other hand, the grid approach is less efficient and requires 1488 beams (due to the higher overlap between beams), effectively assigning a single beam to each user.

For the complete world scenario, our solution covers the 18 712 user terminals with 9633 beams, which corresponds to approximately two users per beam. The reasons behind this low value are rooted in the constellation's characteristics: A low cone aperture angle (4.6°) combined with low orbits results in a very low user-to-user distance threshold (< 40 km). The results, however, are considerably lower than that of the grid approach, reducing the number of beams by 40%.

4.2 | Frequency assignment

Section 4.1 showed that 18 712 users can be grouped into 9633 beams with the approach presented in this work. After the beam placement, the frequency plan needs to be computed. For this purpose, each satellite from Starlink has been assigned a satellite-group (see Section 3.3.1): Satellites in the lower 550-km orbit are all assigned to satellite-group 1. Satellites in the 1110-km altitude orbits are divided into four different satellite-groups, whereas satellites in the 1130, 1275, and 1325 km are each divided into two satellite-groups. In addition, we assume a minimum angular separation of twice the beam size ($\epsilon = 9.2^\circ$).

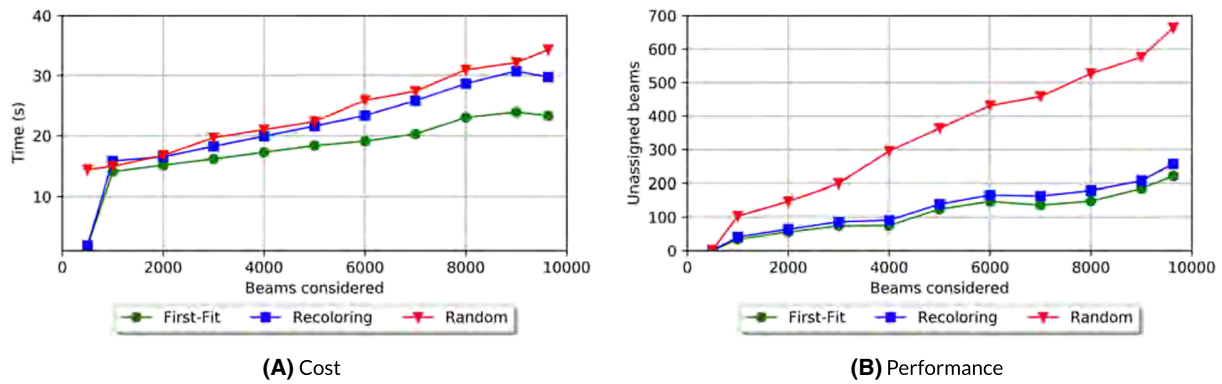


FIGURE 9 Cost and performance of the different frequency assignment algorithms [Colour figure can be viewed at wileyonlinelibrary.com]

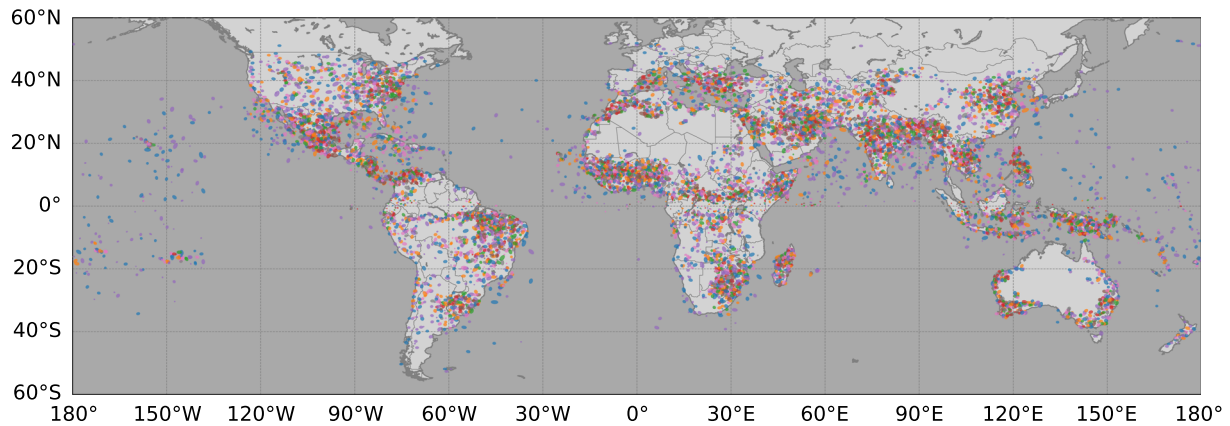


FIGURE 10 Frequency assignment solution for Starlink constellation. Out of the 9633 beams, 9412 were successfully assigned (97.7%) and 221 unassigned [Colour figure can be viewed at wileyonlinelibrary.com]

As done with the beam placement, the scalability of our algorithm is assessed by taking subsets of the beams. Again, we run the algorithm on a single core of a standard laptop.

In this scenario, not all the beams can be assigned a frequency band without violating the constraints. The unassigned beams are near-equatorial beams in dense regions with access to a small number of satellite-groups. Under these conditions, the frequency interference constraints are the limiting factor that prevents the assignment of certain beams.

Figure 9 shows the cost and performance results for the three frequency assignment algorithms considered in this work. In terms of time, all of the algorithms are linear with respect to the number of beams and behave similarly. The random assignment needs a higher computation time for more beams due to the number of trials set (100). On the other hand, the recoloring needs 25% more time than first-fit due to the initialization of all the beams (lines 2 to 5 in Algorithm 2). Regarding performance, the first-fit algorithm outperforms consistently the other methods considered, reaching a reduction of 65% and 10% in the number of unassigned beams over the random assignment and recoloring methods, respectively. Finally, it is worth noting that the number of unassigned beams increases at a lower rate for the first-fit and recoloring (2.5% of the beams are unassigned), while the slope of the curve is steeper for the random assignment (7% of beams unassigned), clearly stating the underperformance of this algorithm.

Figure 10 shows the frequency mapping of the solution found by the algorithm for the whole world. The color of each beam represents the frequency channel it was assigned. As it can be seen, the ocean is filled with blue and purple beams. They have few restrictions, and a high number of channels can be assigned (up to four channels to each isolated ocean beam). In contrast, dense zones have more constraints, so beams can only be assigned one single channel in many cases.

Figure 11 shows the mapping of beams to one satellite over India. The solution shows that the satellite selected is almost at full capacity, using 31/32 of its available channels. To satisfy the interference and frequency reuse constraints, the selected beams are spread out across the satellite's field of view. It can be observed that there are not two beams in close proximity sharing the same frequency, thus satisfying the frequency interference constraints. Since this is a static frequency plan, all users keep their assigned central frequency and bandwidth after switching satellites.

Figure 12 represents the distribution of channels among beams and the spectrum utilization among non-empty satellites. Figure 12A shows that the first and last channels are the most used. This is an expected result, as these two channels are the first to be assigned (since they lead to less constraints).

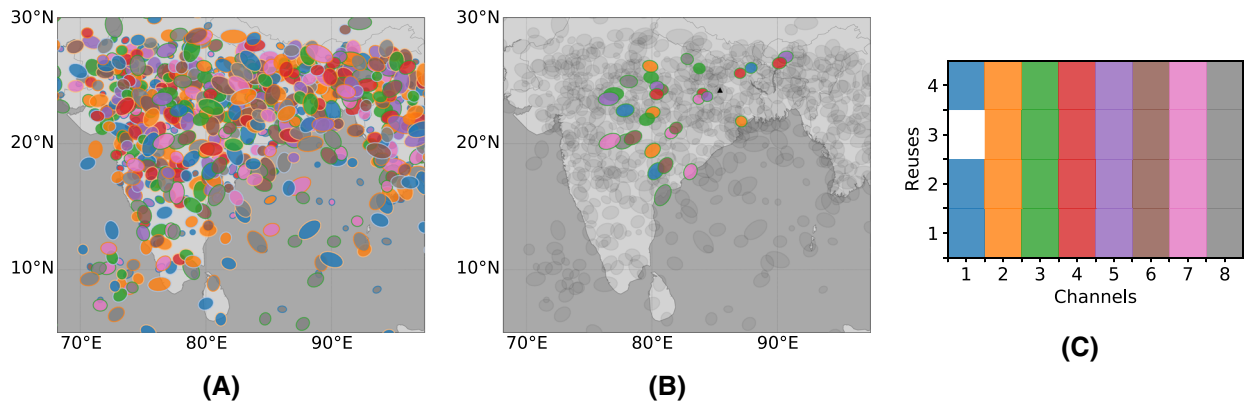
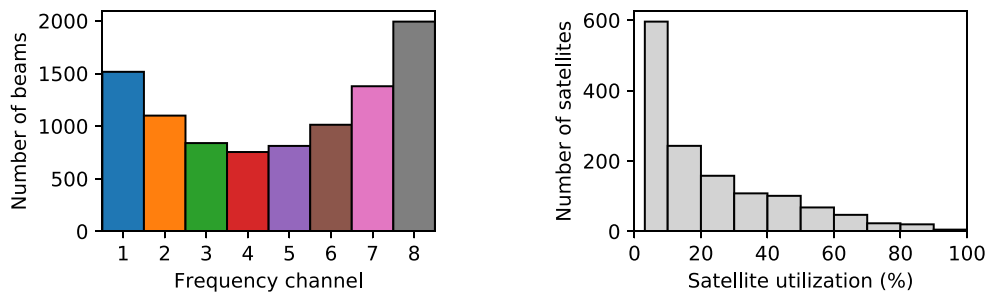


FIGURE 11 Solution of the frequency assignment algorithm over India. A, shows the global assignment for the 737 beams in the region, whereas B, shows the assignment for the beams in a selected satellite. C, depicts the frequency allocation for that satellite. It can be seen that out of the 32 channels available in this satellite, 31 were assigned (96.88%) [Colour figure can be viewed at [wileyonlinelibrary.com](https://onlinelibrary.wiley.com)]



(A) Histogram of channel assignment per beam. **(B)** Histogram of the occupancy of each non-empty satellite.

FIGURE 12 Histograms of channel assignment and satellite occupancy [Colour figure can be viewed at [wileyonlinelibrary.com](https://onlinelibrary.wiley.com)]

Regarding satellite utilization, Figure 12B shows that most satellites are empty or almost empty. From the 4409 satellites that comprise the Starlink constellation, only 1369 appear in the histogram, and almost 600 of them are below a 10% utilization. The reason behind this low utilization lies within constellation characteristics: There are 3358 satellites pointing to empty or almost empty (1 beam) locations in the ocean. This leads to a global low utilization of the constellation.

5 | CONCLUSIONS AND FUTURE WORK

This paper presented an algorithm to solve the beam placement and frequency assignment problems for large LEO constellations. After formulating the problem and describing the algorithm details, we provided results on a real case simulation of SpaceX's constellation with 18 712 users. Finally, we benchmarked our algorithms and compared with other approaches.

The main findings are the following:

- Both the beam placement and the frequency assignment algorithms find solutions within tenth of seconds even in a large scenario with 18 712 users and 9633 beams.
- The beam placement algorithm performs better than a standard grid, as it makes better use of the beam's conical shape. The approach presented is able to reduce the number of beams compared to the grid baseline by 40%. For smaller sets of users, we showed that our algorithm reaches the optimal solution.
- The frequency assignment algorithms (first-fit and recoloring) are able to allocate more than 97% of the beams in the Starlink scenario. Only in very dense areas with few satellites in line of sight, the constraints prevent the assignment of all beams.
- Although originally developed for large LEO constellations, our algorithms can be extended to MEO and GEO constellations with no changes in the formulation.

Future streams of research include adding moving terminals to the formulation, such as airplanes or ships, as currently all users are considered to be fixed. This would introduce new constraints between beams. Recouple the beam placement and frequency assignment and optimize them jointly. Explore dynamic frequency allocation and improve the heuristics used in this work, as a method to reduce the number of unassigned beams and their overall bandwidth allocations.

ACKNOWLEDGMENTS

The authors are grateful to SES S.A. of Betzdorf, Luxembourg, for making available its proprietary (patent pending) Shared Capacity Services (SCS) beam placement technology on which the proposed clique analysis approach is based. The authors would also like to thank SES S.A. for their financial and non financial support in this work. The authors would furthermore like to thank the Higher Education Interdisciplinary Center (CFIS - UPC), in the name of Fundació Cellex, for partially funding this project. Responsibility of the information expressed herein lies entirely with the authors.

AUTHOR CONTRIBUTIONS

Nils Pachler formalized the idea, developed the theory, and performed the computations. Markus Guerster supervised the project and preprocessed the input data. Inigo del Portillo developed the LEO constellation model used in the simulations. All authors contributed to the final version of the manuscript.

ORCID

Nils Pachler de la Osa  <https://orcid.org/0000-0002-3901-7059>

Bruce Cameron  <https://orcid.org/0000-0002-1868-6325>

REFERENCES

1. Ashford EW. Non-Geo systems—where have all the satellites gone? *Acta Astronautica*. 2004;55(3-9):649-657.
2. del Portillo I, Cameron BG, Crawley EF. A technical comparison of three low Earth orbit satellite constellation systems to provide global broadband. *Acta Astronautica*. 2019;159:123-135.
3. Guerster M, Luis JJG, Crawley EF, Cameron BG. Problem representation of dynamic resource allocation for flexible high throughput satellites. In: 2019 IEEE Aerospace Conference IEEE; 2019:1-8.
4. Akkor G, Hadjitheodosiou MS, Baras J. Multicast-aware power allocation in multiple spot-beam satellite communication systems. *Int J Sat Commun Netw*. 2006;24(1):49-71.
5. Aravanis AI, M.R BS, Arapoglou P-D, Danoy G, Cottis PG, Ottersten B. Power allocation in multibeam satellite systems: a two-stage multi-objective optimization. *IEEE Trans Wirel Commun*. 2015;14(6):3171-3182.
6. Garau Luis JJ, Guerster M, del Portillo I, Crawley EF, Cameron BG. Deep reinforcement learning architecture for continuous power allocation in high throughput satellites. arXiv preprint arXiv:190600571; 2019.
7. Pachler N, Garau Luis JJ, Guerster M, Crawley EF, Cameron BG. Allocating power and bandwidth in multibeam satellite systems using particle swarm optimization. In: 2020 IEEE Aerospace Conference IEEE; 2020.
8. Garau Luis JJ, Pachler N, Guerster M, del Portillo I, Crawley EF, Cameron BG. Artificial intelligence algorithms for power allocation in high throughput satellites: a comparison. In: 2020 IEEE aerospace Conference IEEE; 2020.
9. Mizuike T, Ito Y. Optimization of frequency assignment. *IEEE Trans Commun*. 1989;37(10):1031-1041.
10. Orlin J. Contentment in graph theory: covering graphs with cliques. In: *Indagationes mathematicae (proceedings)*, Vol. 80 Elsevier; 1977:406-424.
11. Camino J-T, Artigues C, Houssin L, Mourgues S. Mixed-integer linear programming for multibeam satellite systems design: application to the beam layout optimization. In: 2016 annual IEEE Systems Conference (SysCon) IEEE; 2016:1-6.
12. Camino J-T, Artigues C, Houssin L, Mourgues S. Linearization of Euclidean norm dependent inequalities applied to multibeam satellites design. *Comput Optim Appl*. 2019;73(2):679-705.
13. Kyrgiazos A, Evans B, Thompson P. Irregular beam sizes and non-uniform bandwidth allocation in HTS multibeam satellite systems. In: 31st AIAA International Communications Satellite Systems Conference (ICSSC) AIAA; 2013:5617.
14. Gramm J, Guo J, Hüffner F, Niedermeier R. Data reduction, exact, and heuristic algorithms for clique cover. In: *Proceedings of the meeting on algorithm engineering & experiments. Society for Industrial and Applied Mathematics*; 2006:86-94.
15. Gramm J, Guo J, Hüffner F, Niedermeier R. Data reduction and exact algorithms for clique cover. *J Experimental Algo (JEA)*. 2009;13:2.
16. Wu Q, Hao J-K. A review on algorithms for maximum clique problems. *European J Oper Res*. 2015;242(3):693-709.
17. McIntyre R, Soltys M. An improved upper bound and algorithm for clique covers. *J Discrete Algo*. 2018;48:42-56.
18. Park U, Kim HW, Oh DS, Ku BJ. A dynamic bandwidth allocation scheme for a multi-spot-beam satellite system. *Etri J*. 2012;34(4):613-616.
19. Park U, Kim HW, Oh DS, Ku BJ. Flexible bandwidth allocation scheme based on traffic demands and channel conditions for multi-beam satellite systems. In: 2012 IEEE Vehicular Technology Conference (VTC Fall) IEEE; 2012:1-5.
20. Wang H, Liu A, Pan X, Jia L. Optimal bandwidth allocation for multi-spot-beam satellite communication systems. In: *Proceedings 2013 International Conference on Mechatronic Sciences, Electric Engineering and Computer (MEC) IEEE*; 2013:2794-2798.
21. Pérez Palomar D, Rodríguez Fonollosa J. Practical algorithms for a family of waterfilling solutions. *IEEE Trans Sig Proces*. 2005;25:686-695.
22. Hu X, Liu S, Chen R, Wang W, Wang C. A deep reinforcement learning-based framework for dynamic resource allocation in multibeam satellite systems. *IEEE Commun Lett*. 2018;22(8):1612-1615.
23. Liu S, Hu X, Wang W. Deep reinforcement learning based dynamic channel allocation algorithm in multibeam satellite systems. *IEEE Access*. 2018;6:15733-15742.
24. Funabiki N, Nishikawa S. A gradual neural-network approach for frequency assignment in satellite communication systems. *IEEE Trans Neural Netw*. 1997;8(6):1359-1370.

25. Salcedo-Sanz S, Santiago-Mozos R, Bousoño-Calzón C. A hybrid Hopfield network-simulated annealing approach for frequency assignment in satellite communications systems. *IEEE Trans Syst Man Cyber Part B (Cybern)*. 2004;34(2):1108-1116.
26. Salcedo-Sanz S, Bousoño-Calzón C. A hybrid neural-genetic algorithm for the frequency assignment problem in satellite communications. *Appl Intell*. 2005;22(3):207-217.
27. Cygan M, Pilipczuk M, Pilipczuk M. Known algorithms for edge clique cover are probably optimal. *SIAM J Comput*. 2016;45(1):67-83.
28. SpaceX. Space x non-geostationary satellite system, Space Exploration Holdings LLC. Available at: http://licensing.fcc.gov/myibfs/forwardtopublictabaction.do?file_number=SATMOD2018110800083 (2019/23/08); 2016.

AUTHOR BIOGRAPHIES



Nils Pachler de la Osa is a dual Bachelor Degree student in Aeronautics and Computer Science at the Higher Interdisciplinary Education Centre (CFIS) at the Universitat Politècnica de Catalunya (UPC), Barcelona, Spain. He is currently researching his degree thesis at the AeroAstro Department, Massachusetts Institute of Technology (MIT), under the supervision of Professor Edward Crawley.



Markus Guerster is a 3rd year graduate student pursuing his PhD in the System Architecture Lab of the Department of Aeronautics and Astronautics at MIT. He works on resource allocation for communication satellites with particular focus on a Revenue Management system for operators. He received his MSc degree (2017) with high distinction in Aerospace Engineering as well as his BSc degree (2015) in Mechanical Engineering from the Technical University of Munich.



Inigo del Portillo is a PhD candidate in the Department of Aeronautics and Astronautics at MIT. His research interests include system architectures for spacial and aerial concepts to extend global connectivity, space optical communications systems, and small satellites communications. Inigo received his degrees in Industrial Engineering, Electronics Engineering, and Telecommunications Engineering in 2014 from Universitat Politècnica de Catalunya, Barcelona, and his MSc in Aeronautics and Astronautics from MIT in 2016.



Prof Edward F. Crawley received an ScD in Aerospace Structures from MIT in 1981. His early research interests centered on structural dynamics, aeroelasticity, and the development of actively controlled and intelligent structures. Recently, Dr Crawley's research has focused on the domain of the architecture and design of complex systems. From 1996 to 2003, he served as the Department Head of Aeronautics and Astronautics at MIT, leading the strategic realignment of the department. Dr Crawley is a Fellow of the AIAA and the Royal Aeronautical Society (UK) and is a member of three national academies of engineering. He is the author of numerous journal publications in the *AIAA Journal*, the *ASME Journal*, the *Journal of Composite Materials*, and *Acta Astronautica*. He received the NASA Public Service Medal. Recently, Prof Crawley was one of the 10 members of the presidential committee led by Norman Augustine to study the future of human spaceflight in the United States.



Dr Bruce Cameron is a Lecturer in Engineering Systems at MIT and a consultant on platform strategies. At MIT, Dr Cameron ran the MIT Commonality study, a 16-firm investigation of platforming returns. Dr Cameron's current clients include Fortune 500 firms in high tech, aerospace, transportation, and consumer goods. Prior to MIT, Bruce worked as an engagement manager at a management consultancy and as a system engineer at MDA Space Systems and has built hardware currently in orbit. Dr Cameron received his undergraduate degree from the University of Toronto and graduate degrees from MIT.

How to cite this article: Pachler de la Osa N, Guerster M, del Portillo Barrios I, Crawley E, Cameron B. Static beam placement and frequency plan algorithms for LEO constellations. *Int J Satell Commun Network*. 2020;1-13. <https://doi.org/10.1002/sat.1345>