

Fast Beam Placement for Ultra-Dense LEO Networks

Trinh Van Chien, Nguyen Minh Quan, Tri Nhu Do, Cuong Le, Tan N. Nguyen, and Symeon Chatzinotas

Abstract—Low Earth orbit (LEO) satellites has brought about significant improvements in wireless communications, characterized by low latency and reduced transmission loss compared to geostationary orbit (GSO) satellites. Ultra-dense LEO satellites can serve many users by generating active beams effective to their locations. The beam placement problem is challenging but important for efficiently allocating resources with a large number of users. This paper formulates and solves a fast beam placement optimization problem for ultra-dense satellite systems to enhance the link budget with a minimum number of active beams (NABs). To achieve this goal and balance load among beams within polynomial time, we propose two algorithms for large user groups exploiting the modified K-means clustering and the graph theory. Numerical results illustrate the effectiveness of the proposals in terms of the statistical channel gain-to-noise ratio and computation time over state-of-the-art benchmarks.

Index Terms—Beam placement, clustering, minimum clique cover, half power beam width.

I. INTRODUCTION

Non-geostationary orbit (Non-GSO) satellites such as low Earth orbit (LEO) have boosted the system performance by offering non-terrestrial connectivity solutions and supporting diverse digital technologies [1]. These satellites have the potential to provide low latency, enhance communication speeds, and improve energy efficiency. Accordingly, there are plans to launch tens of thousands of Non-GSO satellites to establish ultra-dense mega-constellations [2]. This will result in an opportunity to cover the Earth, characterized by a variety of frequencies and directions. Besides, resource allocation is a critical issue in satellite communications. There are remaining challenges in enhancing the efficiency of limited resources and increasing satellite throughput due to the constraints on the limited budgets. The primary objectives involve minimizing costs, reducing computation time, and controlling congestion [3]. An optimization problem may entail a vast number of integer variables within a discrete search space, thereby resulting in its worst-case complexity being NP-hard [4]. In this context, we address a beam placement problem, where conical beams are defined by their half-power beam width (HPBW). Our aim is to identify suitable locations within the coverage area to position these beams, optimizing their

arrangement to maximize the number of users per satellite beam and determining the optimal beam center.

Recent studies have explored various approaches to the beam placement for LEO constellations. The authors in [5] utilized the gradient ascent and quadratic transform for an iterative-based approach to determine the beam center and transmit power. Pachler *et al.* employed the graph theory and a heuristic algorithm to place the beams [6]. However, these works have not specified the active beam number or addressed the user distribution disparity among beams. This can lead to load imbalance, reducing communication efficiency. Some proposed approaches to tackle the load balancing problem include linear optimization techniques, self-organizing feature maps, and problem linearizations. Nevertheless, these methods have not classified users into individual beams and specified the minimal number of active beams. The recently proposed methods including the deterministic annealing in [7] and the two-stage algorithm based K-means clustering in [8] can ensure balanced traffic load among employed beams. However, they only considered a small number of users with the time consumption being inadequate for practical applications.

This work suggests strategies to minimize the number of active beams and improve the load balance in the network serving many users. We address the optimization process of reducing the distance between users and their corresponding serving beams. This particular topic has not been thoroughly investigated in this literature because of the non-convex and NP-hard properties. We concentrate on developing two algorithms to cluster users based on the Half Power Beam Width (HPBW) to obtain a solution in a polynomial time with the following key contributions: *i)* we develop a system where multi-beams serve multi-users as well as formulating an optimization problem; *ii)* we propose two algorithms: one utilizing loops in conjunction with K-means clustering, and another employing a two-phase algorithm based on graph theory; and *iii)* we provide experimental results to compare the effectiveness of the two proposed algorithms and compare with the algorithms provided in the document [8]. Specifically, the proposed algorithms can be applied for large-scale networks with many users.

II. ULTRA-DENSE SATELLITE COMMUNICATION & BEAM PLACEMENT OPTIMIZATION PROBLEM

This section presents a high throughput multi-beam system. A beam placement problem is then formulated and solved.

A. System Model

A satellite is equipped with N beams having the circle pattern to serve K users, where multiple users can be served by one beam. Due to energy limitations and mutual interference,

T. V. Chien and N. M. Quan are with the School of Information and Communication Technology, Hanoi University of Science and Technology, Hanoi 100000, Vietnam (e-mail: chientv@soict.hust.edu.vn and quan.nm200508@sis.hust.edu.vn). T. N. Do is with the Department of Electrical Engineering, Polytechnique Montreal, Montreal, QC H3T 1J4, Canada (e-mail: tri-nhu.do@polymtl.ca). T. N. Nguyen is with the Communication and Signal Processing Research Group, Faculty of Electrical and Electronics Engineering, Ton Duc Thang University, Ho Chi Minh City, Vietnam (e-mail: nguyennhathan@tdtu.edu.vn). C. Le and S. Chatzinotas are with the with the Interdisciplinary Centre for Security, Reliability and Trust (SnT), University of Luxembourg, 1855 Luxembourg, Luxembourg (e-mail: vancuong.le@uni.lu and symeon.chatzinotas@uni.lu). Corresponding author: Tan N. Nguyen.

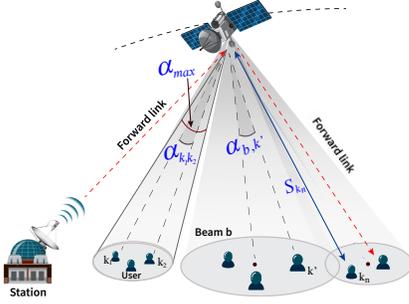


Fig. 1: Illustration of the system model, where multiple users can be served by the same beams.

it might not be efficient to activate all the beams. Let B and $\mathcal{B} = \{1, 2, \dots, B\}$ be the number and the set of active beams, $\mathcal{U} = \{k_1, k_2, \dots, k_K\}$ be the set of all users, and $\mathcal{U}_b \subseteq \mathcal{U}$ be a subset of users served by the b -th active beam. Then the beam placement problem is defined as assigning the set of users \mathcal{U} to a subset \mathcal{B} such that the NABs are minimized, where every user must be served by exactly one beam. The practical constraints are expressed as

$$\bigcup_{b \in \mathcal{B}} \mathcal{U}_b = \mathcal{U}; \mathcal{U}_b \neq \emptyset, \mathcal{U}_b \cap \mathcal{U}_{b'} = \emptyset, \forall b, b' \in \mathcal{B}, b \neq b'. \quad (1)$$

We assume that a user is served by a given beam if and only if this user is located in at least the HPBW of the beam. Let $\alpha_{b,k} \in [0, \pi/2]$ be the angle between user k and the centre of beam b (see Fig. 1), and g_{bk} be the radio pattern between beam b and user k . Also, g_{\max} is the maximum gain obtained when user k is located at the beam centre. Then, user k can be served by beam b if and only if $g_{bk} \geq g_{\max}/2$. According to the 3GPP report [9], $g_{bk} = g_{\max}G(\alpha_{b,k}), \forall b \in \mathcal{B}, k \in \mathcal{U}$, where $G(\alpha_{b,k})$ is the normalized pattern gain, which is

$$G(\alpha_{b,k}) = \begin{cases} 1, & \text{if } \alpha_{b,k} = 0, \\ 4 \left| \frac{J_1\left(\frac{2\pi}{\lambda} r \sin(\alpha_{b,k})\right)}{\frac{2\pi}{\lambda} r \sin(\alpha_{b,k})} \right|^2, & \text{if } 0 < \alpha_{b,k} \leq \frac{\pi}{2}, \end{cases} \quad (2)$$

where $J_1(\cdot)$ is the Bessel function of the order one, λ is the carrier wavelength, and r is the radius of the antenna aperture. Let ϵ_{bk} be the antenna efficiency and S_k be the slant range distance between user k and the satellite. D is the satellite antenna diameter. The statistical channel gain (SCG) at an arbitrary user is $G_{bk} = g_{bk}\bar{g}_{bk}/(L_{bk}L_{atm})$, where $\bar{g}_{bk} = \epsilon_{bk}\pi^2 D^2/\lambda^2$ is the received antenna power gain, $L_{bk} = 16\pi^2 S_k^2/\lambda^2$ is the free path loss between user k and the satellite, and L_{atm} is the atmospheric absorption. The statistical channel gain-to-noise ratio (SCGNR) at an user k is denoted as $SCGNR_k = G_{bk}/\sigma^2$, where σ^2 is the noise power.

B. Problem Formulation

Let's define decision variables $x_b = \{0, 1, \forall b \in [1, N]\}$ where $x_b = 1$ as beam b is active. Otherwise, this beam is inactive. Binary variables $y_{bk} = \{0, 1, \forall b \in [1, N], k \in \mathcal{U}\}$ are introduced to assign users to the beams. Specifically, $y_{bk} = 1$ if user k is assigned to beam b , and $y_{bk} = 0$ otherwise. The beam placement problem now becomes finding

$\mathbf{x} = \{x_b, \forall b \in [1, N]\}$ and $\mathbf{y} = \{y_{bk}, \forall b \in [1, N], k \in \mathcal{U}\}$ such that the NABs is minimized. The problem is formulated as

$$\underset{\mathbf{x}, \mathbf{y}}{\text{minimize}} \quad \sum_{b=1}^N x_b \quad (\text{P})$$

$$\text{subject to} \quad \sum_{b=1}^N y_{bk} = 1, \forall k \in \mathcal{U}, \quad (3a)$$

$$x_b \leq \sum_{k=1}^K y_{bk}, \forall b \in [1, N], \quad (3b)$$

$$M(\beta_{bk} - 1) \leq g_{bk} - g_{\max}/2 \leq M\beta_{bk}, \\ \forall b \in [1, N], k \in \mathcal{U}, \quad (3c)$$

$$y_{bk} \leq \beta_{bk}, \forall b \in [1, N], k \in \mathcal{U}, \quad (3d)$$

$$x_b, y_{bk}, \beta_{bk} \in \{0, 1\}, \forall b \in [1, N], k \in \mathcal{U}, \quad (3e)$$

where M is a large constant and β_{bk} are auxiliary variables to represent whether the HPBW condition is satisfied, i.e., $\beta_{bk} = 1$ implies that $g_{bk} \geq g_{\max}/2$ and user k can be served by the b -th beam, and $\beta_{bk} = 0$ otherwise. The objective of problem P is to minimize the NABs. Constraint (3a) is to ensure that all the users are served and each of those is served by exactly one beam. Constraint (3b) guarantees that no beam is activated without serving any users. Constraint (3c) is to linearize the HPBW condition. Constraint (3d) ensures that all users are within the coverage regions of the assigned beams.

C. Connection (P) with the Minimum Clique Cover Problem

In this section, we connect our optimization problem to the well-known graph theory problem, Minimum Clique Cover, which involves partitioning the vertex set into the smallest number of disjoint cliques. Let's define an undirected graph $\mathbf{G} = (V, E)$, where $V = \mathcal{U}$ is a set of vertices and $E = \{(k_i, k_u), k_i, k_u \in V\}$ is a set of edges. Each vertex $k \in V$ corresponds to user k in \mathcal{U} , and each edge $e = (k_i, k_u)$ indicates that two users k_i and k_u can be served by a single beam. We consider the HPBW condition to determine whether two users can be served by the same beam. Let $\alpha_{k_i k_u}$ be the angle from the satellite to user k_i and k_u ($k_i, k_u \in \mathcal{U}$), calculated as $\alpha_{k_i k_u} = \arccos(2S_{k_i} S_{k_u} / (S_{k_i}^2 + S_{k_u}^2 - d_{k_i, k_u}^2))$. Where, d_{k_i, k_u} is the geography distance, computed through the Spherical Law of Cosines. We note that α_{\max} is the HPBW of beam b , $\forall b \in \mathcal{B}$, i.e., the normalized pattern gain $G(\alpha_{\max}/2) = 1/2$. Then two users k_i and k_u can be served by the same beam b if and only if $\alpha_{k_i k_u} \leq \alpha_{\max}$ (see Fig. 1). However, [6] showed that some users cannot be covered if using an HPBW of α_{\max} . Thus, we use $\alpha_{\max}/2$ to ensure that all users in the same beam are within the HPBW area.

Then the graph \mathbf{G} can be represented by an adjacency matrix \mathbf{A} defined as $\mathbf{a}_{k_i k_u} = 1$, if $\alpha_{k_i k_u} \leq \alpha_{\max}/2$, and $\mathbf{a}_{k_i k_u} = 0$, otherwise. Therefore, to ensure that all users belong to a single beam b_t , the following condition must be satisfied: $\mathbf{a}_{k_i k_u} = 1, \forall k_i, k_u \in b_t$. This means that each pair of vertices corresponding to users who belong to the same beam is connected. Thus, a subset of V corresponding to the subset of users that satisfies Problem (P) constitutes a clique, i.e., a complete sub-graph of \mathbf{G} . Our problem is now equivalent to partitioning the vertex set V into B disjoint subsets V_1, V_2, \dots, V_B , where B is minimized and the sub-graph induced by V_b is a complete graph, $\forall 1 \leq b \leq B$. This problem

is known as the Minimum Clique Cover (MCC) problem, which is NP-hard [10]. For a network of K users with a non-limited NABs, it is also NP-hard to approximate Problem (P) within an approximation ratio of $K^{1-\epsilon}$ for any given $\epsilon > 0$.

III. BEAM PLACEMENT WITH BISECTION K-MEANS

This section proposes an algorithm that combines K-means clustering and binary search, namely Bisection K-means (BK-Means) as shown in Algorithm 1 to tackle Problem (P). The primary objective of the clustering algorithm is to divide a given dataset into M groups, where the value of M is predetermined. Additionally, we use Bisection algorithm to explore different scenarios of the necessary NABs. Given that $(\phi_{k_i}, \theta_{k_i})$ is longitude and latitude of user k_i and R is the radius of the Earth. We utilize Cartesian coordinate format denoted by $(x_{k_i}, y_{k_i}, z_{k_i})$ as: $x_{k_i} = R \cos(\phi_{k_i}) \cos(\theta_{k_i})$, $y_{k_i} = R \cos(\phi_{k_i}) \sin(\theta_{k_i})$, $z_{k_i} = R \sin(\phi_{k_i})$, $\forall i \in [1, K]$ to recalculate the distance between k_i and k_u ($\forall k_i, k_u \in \mathcal{U}$) which is employed in K-means clustering as $d_{k_i, k_u} = \sqrt{(x_{k_i} - x_{k_u})^2 + (y_{k_i} - y_{k_u})^2 + (z_{k_i} - z_{k_u})^2}$. The algorithm involves three main steps: *i*) determining the number of clusters B for testing; *ii*) performing K-means clustering with B clusters repeatedly until a solution is found or a maximum of μ iterations is reached; and *iii*) repeating steps *i*) and *ii*) until the termination condition is met. From set of N beams, we divide range of beams into 2 ranges with a mean value B_{mean} , this value is used to test the feasibility of NABs, and then making a decision to resize the range accordingly. The NABs can be in $[1; B_{\text{mean}}]$ only if a solution with B_{mean} beams is feasible. Otherwise, it falls within the range of $[B_{\text{mean}} + 1; N]$. To evaluate the feasibility of B_{mean} beams, we utilize the adjacency matrix \mathbf{A} of graph \mathbf{G} . We denote $\mathcal{S} = \{\mathcal{U}_1, \mathcal{U}_2, \dots, \mathcal{U}_{B_{\text{mean}}}\}$, $\mathcal{U}_i \in \mathcal{S}$, $i \in \{1, 2, \dots, B_{\text{mean}}\}$ as a solution of problem (P), so feasible can be concluded as

$$\text{feasible} = \begin{cases} \text{False,} & \text{if } \exists (k_i, k_u) \in \mathcal{U}_i \text{ s.t } \mathbf{a}_{k_i k_u} = 0 \\ \text{True,} & \text{otherwise.} \end{cases} \quad (4)$$

Computational complexity: Time complexity of BK-Means is estimated as $O(\mu KN \log(N)(NI + K))$, with N maximum number of beams, K users, I iterations of K-means clustering, and μ maximum number of iterations.

IV. GRAPH COLORING BASED BEAM PLACEMENT

This section proposes a two-phase algorithm: first, it solves MCC problem to get an initial solution, then iteratively refines this solution for better load balancing among active beams.

Phase 1 (solution initialization): Our approach is transforming the MCC problem into the equivalent Graph Coloring (GC) problem [10]. The GC problem involves assigning colors to the vertices of the graph in such a way that no two adjacency nodes have the same color, and the number of colors is minimized. Let $\bar{\mathbf{G}} = (V, \bar{E})$ be the complement graph of \mathbf{G} . That is, $\bar{\mathbf{G}}$ is constructed on the same vertex set of \mathbf{G} and two vertices of $\bar{\mathbf{G}}$ is connected if and only if they are independent in \mathbf{G} . Let $\bar{\mathbf{A}}$ be the adjacency matrix of $\bar{\mathbf{G}}$ given as:

$$\bar{a}_{k_i k_u} = \begin{cases} 1, & \text{if } a_{k_i k_u} = 0, \\ 0, & \text{otherwise.} \end{cases} \quad (5)$$

Algorithm 1 Bisection K-Means clustering based beam placement (BK-Means)

- 1: **Input:** Dataset, graph \mathbf{G} , matrix \mathbf{A} , number of maximum beams N , maximum iterations μ .
 - 2: **Output:** List of active beams \mathcal{B} and user assignment \mathcal{S} .
 - 3: Initialize $B_{\text{min}} = 1$, $B_{\text{max}} = N$;
 - 4: **while** $B_{\text{min}} + 1 < B_{\text{max}}$ **do**
 - 5: Set $B_{\text{mean}} = \lfloor (B_{\text{min}} + B_{\text{max}})/2 \rfloor$;
 - 6: feasible = False; count = 0;
 - 7: **while** not feasible and count $< \mu$ **do**
 - 8: Run K-means with B_{mean} clusters;
 - 9: **if** every cluster forms a clique of \mathbf{G} **then**
 - 10: feasible = True;
 - 11: **else**
 - 12: count = count + 1;
 - 13: **if** feasible **then**
 - 14: $B_{\text{max}} = B_{\text{mean}}$;
 - 15: **else**
 - 16: $B_{\text{min}} = B_{\text{mean}}$;
 - 17: **Return:** B subsets of users corresponding to B beams;
-

Finding minimum clique cover of \mathbf{G} is equivalent to finding a proper vertex coloring for $\bar{\mathbf{G}}$ such that the number of colors is minimized. Each subset of vertices in $\bar{\mathbf{G}}$ that shares the same color as an active beam in our problem. To solve this problem, we use a greedy algorithm which starts with a list of vertices of $\bar{\mathbf{G}}$ sorted in descending order of their degrees. The algorithm iteratively assigns the smallest color to a vertex and only uses a new color if all previously used colors are already assigned to its neighbors, ensuring that adjacency vertices have different colors.

Phase 2 (load balancing improvement): The initial solution is iteratively refined by moving users from high-workload beams to lower-workload beams. The algorithm considers all the beam pairs, checks and re-assigns unbalanced assignments. The process is repeated until no further feasible reassignments are possible. More details of these steps are described in Phase 2 of Algorithm 2 with the convergence as follows.

Lemma 1. *Phase 2 of Algorithm 2 converges after BK movements, with the NABs (B) and the number of users (K).*

Proof. Consider $\mathcal{S} = \{\mathcal{U}_1, \mathcal{U}_2, \dots, \mathcal{U}_B\}$ being an result of Phase 1. Let $u_b = |\mathcal{U}_b|$, $\forall \mathcal{U}_b \in \mathcal{S}$. Assuming that the set \mathcal{S} is sorted in the descending order of its elements' sizes, i.e., $u_1 \geq u_2 \geq \dots \geq u_B$. Let define $\Phi_b = \sum_{j=b+1}^B (u_b - u_j)$ be the balancing indicator of beam b and $\Phi = \sum_{b=1}^B \Phi_b$ be the balancing indicator of the whole system. We have

$$\Phi = \sum_{b=1}^B \sum_{j=b+1}^B (u_b - u_j) = \sum_{b=1}^B ((B-b)u_b - \sum_{j=b+1}^B u_j) \leq B \sum_{b=1}^B u_b = B \sum_{b=1}^B |\mathcal{U}_b| = BK. \quad (6)$$

Since we only move a user from group \mathcal{U}_b to group $\mathcal{U}_{b'}$, if $|\mathcal{U}_b| - |\mathcal{U}_{b'}| > 1$, the balancing indicator Φ decreases by at least one after each movement. Consequently, Algorithm 2 finishes within a maximum of BK steps. \square

Algorithm 2 Two-phase Greedy Beam Placement (TGBP)

```

1: Input: List of users  $\mathcal{U}$ , graph  $\overline{\mathbf{G}}$  and matrix  $\overline{\mathbf{A}}$ .
2: Output: List of active beams  $\mathcal{B}$  and user assignment  $\mathcal{S}$ .
   % Phase 1: Solution initialization
3: Sort all users  $\mathcal{U}$  in descending order of degrees in  $\overline{\mathbf{G}}$ .
4: Initialize  $\mathcal{B} = \emptyset$ , index  $b = 0$ , user assignment  $\mathcal{S} = \emptyset$ ;
5: foreach user  $k$  in the sorted list do
6:   if  $k$  is not covered by any beam then
7:      $\mathcal{U}_b = \{k\}$ ; Mark user  $k$  as covered;
8:     foreach user  $u$  after  $k$  in the sorted list do
9:       if  $u$  is not covered and  $\overline{\mathbf{a}}_{uu'} = 0 \forall u' \in \mathcal{U}_b$  then
10:         $\mathcal{U}_b = \mathcal{U}_b \cup \{u\}$ ; Mark user  $u$  as covered;
11:     $\mathcal{S} = \mathcal{S} \cup \{\mathcal{U}_b\}$ ;  $b = b + 1$ ;
12: Return:  $\mathcal{B}$ ,  $\mathcal{S}$ ;
13: % Phase 2: Load balancing improvement
14: stop = False;
15: while not stop do
16:   count_moves = 0;
17:   foreach pair of user groups  $\mathcal{U}_b, \mathcal{U}_{b'} \in \mathcal{S}$  do
18:     if the load balancing gap  $|\mathcal{U}_b| - |\mathcal{U}_{b'}| > 1$  then
19:       foreach user  $k$  in  $\mathcal{U}_b$  do
20:         if  $\mathbf{a}_{kk'} = 1 \forall k' \in \mathcal{U}_{b'}$  then
21:           Move user  $k$  from  $\mathcal{U}_b$  to  $\mathcal{U}_{b'}$ ;
22:           count_moves = count_moves + 1;
23:         if  $|\mathcal{U}_b| - |\mathcal{U}_{b'}| \leq 1$  then
24:           break;
25:   if count_moves = 0 then
26:     stop = True;
27: Return:  $\mathcal{B}$ ,  $\mathcal{S}$ ;

```

Computational complexity: The complexity of Algorithm 2 is $\mathcal{O}(N^3 K^2)$ with K users and N maximum number of beams. Overall, we provide the performance guarantee for the solutions delivered by Algorithm 2 as in Lemma 2.

Lemma 2. *The NABs (\mathcal{B}) returned by Algorithm 2 is bounded by $B \leq K + 1 - \delta(\mathbf{G})$, where $\delta(\mathbf{G}) = \min \{\deg_{\mathbf{G}}(v) : v \in V\}$ is the minimum degree over all vertices of the graph \mathbf{G} .*

Proof. Since the number of user groups keeps unchanged in Phase 2, the NABs returned by Algorithm 2 only depends on Phase 1. Let $\Delta(\overline{\mathbf{G}}) = \max \{\deg_{\overline{\mathbf{G}}}(v) : v \in V\}$ be the maximum degree over all vertices. Following [11], the list of active beams returned by Phase 1 satisfying that $|\mathcal{B}| \leq \Delta(\overline{\mathbf{G}}) + 1$. Moreover, since $\overline{\mathbf{G}}$ is the complement graph of \mathbf{G} , we have $\deg_{\overline{\mathbf{G}}}(v) = |V| - \deg_{\mathbf{G}}(v), \forall v \in V$. In summary, we have:

$$\begin{aligned}
 |\mathcal{B}| &\leq \max \{\deg_{\overline{\mathbf{G}}}(v) : v \in V\} + 1 \\
 &= \max \{|V| - \deg_{\mathbf{G}}(v) : v \in V\} + 1 \\
 &= |V| + 1 + \max \{-\deg_{\mathbf{G}}(v) : v \in V\} \\
 &= K + 1 - \min \{\deg_{\mathbf{G}}(v) : v \in V\},
 \end{aligned}$$

which completes the proof. \square

V. NUMERICAL RESULTS

We investigate a LEO satellite system providing simultaneous service to multiple users, which are located in a longitude

range of $[30^\circ, 40^\circ]$ and a latitude range of $[-120^\circ, -110^\circ]$. The satellite is positioned at coordinates with the latitude and longitude of $[0^\circ, -88.7^\circ]$, and the satellite antenna has a diameter of 0.6 [m]. Each beam center of the satellite has a maximum gain of 50 [dBi] and a HPBW with a maximum angle of $\alpha_{\max} = 3.2^\circ$. The carrier frequency is 18.05 [GHz], and the aperture radius is 5λ . The maximum iteration is $\mu = 200$, and the maximum iteration number of K-means in Algorithm 1 (line 6) is $I = 500$. All the benchmarks are implemented in Matlab with the same configurations: Windows 11 Pro operating system, Intel Core i7 – 12700 processor, 2.10GHz CPU, and 64GB RAM. We conducted experiments to compare the performance of four algorithms: a heuristic method based on graph theory in [6], referred to as HCBP; K-means with 2 phases (TPK-Means) in [8], BK-Means, and TGBP. We note that TPK-Means and HCBP from previous studies have a significant limitation due to their high computational time, making them unsuitable for large user groups. Therefore, to compare the performance of our two algorithms with TPK-Means and HCBP, we consider two cases: small user sets (10-30 users) and large user sets (50-1000 users), with TPK-Means and HCBP only applicable to small user sets.

Fig. 2(a) and Fig. 2(b) show the average NABs required for power distribution using HCBP, TPK-Means, BK-Means, and TGBP. In small user sets (Fig. 2(a)), the NABs are different across algorithms, TGBP is more efficient overall. For larger user sets (Fig. 2(b)), TGBP outperforms BK-Means as the user count increases. This is due to TGBP's Greedy Algorithm, which finds the minimum NABs sequentially, unlike the iterative K-Means-based approaches of BK-Means and TPK-Means. HCBP's incremental algorithm, which selects random cliques by size, also struggles to guarantee optimal solutions.

Fig. 2(c) and Fig. 2(d) depict the average load balancing gap, which is the average difference between the largest beam (serving the most users) and the smallest beam (serving the least users) resulting from all algorithms across various user sets. In Fig. 2(c), there is a noticeable difference among the four algorithms as the user count increases. Specifically, TGBP proves to offer a more balanced distribution compared to BK-Means, TPK-Means and HCBP. This performance is due to the refining phase of TGBP, which enhances balance in smaller user sets through multiple refining steps. However, with larger user sets (500 and 1000 users), BK-Means demonstrates a better average load balancing gap compared to TGBP, as illustrated in Fig. 2(d). This shows that the balancing efficiency of TGBP decreases as the number of refining steps increases with larger user sets. Conversely, BK-Means benefits from its iterative clustering approach, which better manages larger datasets and maintains a more consistent load balancing gap.

Fig. 3(a) and Fig. 3(b) show the cumulative distribution function (CDF) of the statistical channel gain to noise ratio (SCGNR) [dB] for each user. *i.e.*, $10 \log_{10}(G_{b,k}/\sigma_k^2), \forall b \in \mathcal{B}, k \in \mathcal{U}$. BK-Means provides benefits in both average and minimum SCGNR compared to the remaining algorithms in small sets and TGBP in large sets. The channel gain from applying TGBP always less than HCBP, TPK-Means and BK-Means. This is because TGBP prioritizes minimizing the

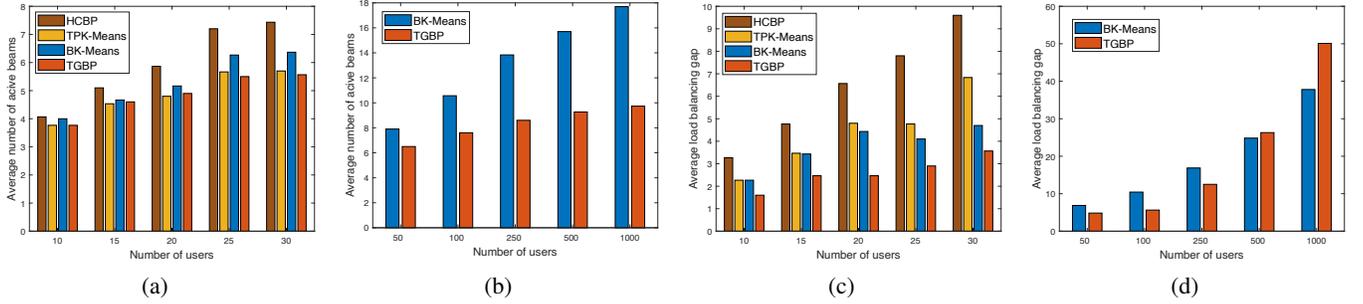


Fig. 2: (a) Number of active beams on small-scale networks. (b) Number of active beams on large-scale networks. (c) Average load balancing gap on small-scale networks. (d) Average load balancing gap on large-scale networks.

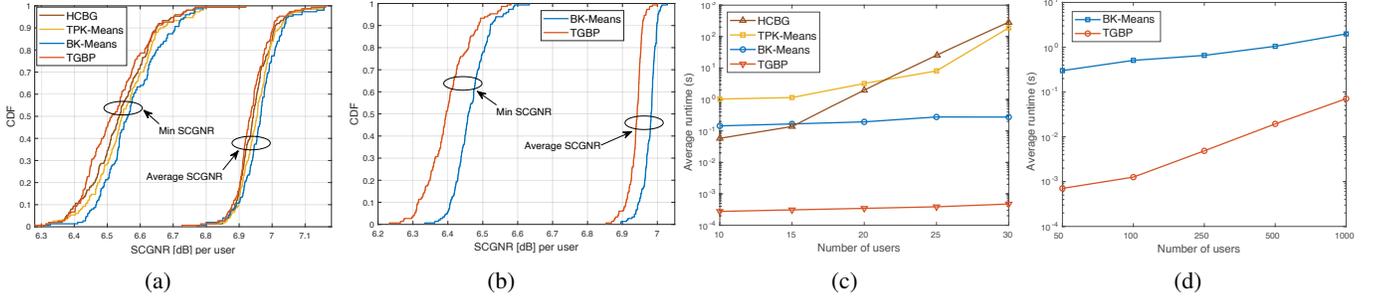


Fig. 3: (a) CDF of the average SCGNR and min SCGNR [dB] per user on small-scale networks. (b) CDF of the average SCGNR and min SCGNR [dB] per user on large-scale networks. (c) Computing time required on small-scale networks. (d) Computing time required on large-scale networks.

number of beams and balancing the load across beams. As a result, each beam in TGBP can typically serve a larger number of users compared to the other algorithms, leading to decreased channel gain per user. BK-Means, with its iterative clustering approach, can better optimize beam placement and power allocation, resulting in higher SCGNR values. TPK-Means and HCBP also benefit from their approaches, but their high computational time makes them less practical for larger user sets despite their potential for achieving higher SCGNR in smaller sets.

Fig. 3(c) and Fig. 3(d) illustrate the running time of each algorithm. TGBP algorithm has a shorter computational time compared to the remaining algorithms, i.e., in the order of milliseconds. Despite a rapid increase in computational time with a rising number of users (from 100 to 1000) leading to the large number of refining steps, TGBP consistently outperforms BK-Means in terms of running time. On the other hand, BK-Means exhibits a gradual and stable increase in computational time due to its iterative K-Means clustering process with a predefined number of iterations. In contrast, HCBP and TPK-Means experience an exponential rise in computational time, making them impractical for large datasets.

VI. CONCLUSION

This paper has illustrated the importance of optimizing beam placement to enhance channel gain for users in the coverage area. Furthermore, load balancing among active beams also helps to save resources while improving communication performance. We have proposed two algorithms to improve computational ability and achieve good feasible solutions in polynomial time compared to the state-of-the-art benchmarks. Numerical experiments illustrated the benefits of two our proposed approaches on channel gain and computation time.

REFERENCES

- [1] T. Darwish, G. K. Kurt, H. Yanikomeroglu, M. Bellemare, and G. Lantagne, "LEO Satellites in 5G and Beyond Networks: A Review From a Standardization Perspective," *IEEE Access*, vol. 10, pp. 35 040–35 060, 2022.
- [2] S. Kota and G. Giambene, "6G Integrated Non-Terrestrial Networks: Emerging Technologies and Challenges," in *2021 IEEE International Conference on Communications Workshops*, 2021, pp. 1–6.
- [3] G. Saravana Kumar, K. K. Ramachandran, S. Sharma, R. Ramesh, K. Qureshi, and K. Ganesh, "AI-Assisted Resource Allocation for Improved Business Efficiency and Profitability," in *2023 3rd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, 2023, pp. 54–58.
- [4] D. U. Kim, S. B. Park, C. S. Hong, and E. N. Huh, "Resource Allocation and User Association Using Reinforcement Learning via Curriculum in a Wireless Network with High User Mobility," in *2023 International Conference on Information Networking (ICOIN)*, 2023, pp. 382–386.
- [5] H.-H. Choi, G. Park, K. Heo, and K. Lee, "Joint Optimization of Beam Placement and Transmit Power for Multibeam LEO Satellite Communication Systems," *IEEE Internet of Things Journal*, vol. 11, no. 8, pp. 14 804–14 813, 2024.
- [6] N. Pachler de la Osa, M. Guerster, I. del Portillo Barrios, E. Crawley, and B. Cameron, "Static beam placement and frequency plan algorithms for LEO constellations," *International Journal of Satellite Communications and Networking*, vol. 39, no. 1, pp. 65–77, 2021.
- [7] N. Torkzaban, A. Zoukarni, A. Gholami, and J. S. Baras, "Capacitated Beam Placement for Multi-beam Non-Geostationary Satellite Systems," in *2023 IEEE Wireless Communications and Networking Conference (WCNC)*, 2023, pp. 1–6.
- [8] V.-P. Bui, T. Van Chien, E. Lagunas, J. Grotz, S. Chatzinotas, and B. Ottersten, "Joint Beam Placement and Load Balancing Optimization for Non-Geostationary Satellite Systems," in *2022 IEEE International Mediterranean Conference on Communications and Networking (Med-ItCom)*, 2022, pp. 316–321.
- [9] 3GPP, "Study on New Radio (NR) to support non-terrestrial networks," *TR 38.811 V15. 2.0*, 2019.
- [10] R. Karp, *Reducibility among combinatorial problems*. Springer, 2010.
- [11] S. Arumugam, A. Brandstädt, T. Nishizeki *et al.*, *Handbook of graph theory, combinatorial optimization, and algorithms*. CRC Press, 2016, vol. 34.