ADAPTIVE MULTIBEAM HOPPING IN GEO SATELLITE NETWORKS WITH NON-UNIFORMLY DISTRIBUTED GROUND USERS

Heba Shehata[1], Hazer Inaltekin[1], Iain B. Collings[1]

[1]Faculty of Science and Engineering, Macquarie University, Sydney, Australia

NOTE: Corresponding author: Heba Shehata, heba.shehata@students.mq.edu.au

Abstract – *This paper designs a novel low-complexity user-cluster grouping algorithm for adaptive beam hopping in geo-stationary satellite networks equipped with multibeam phased-array antennas. Each beam serves a cluster of users, and the challenge is to design a beam-hopping pattern where no beams are simultaneously serving nearby user clusters. We develop a line search procedure to identify near-optimum groupings for heterogeneous traffic demands. We provide a necessary condition to determine the boundaries of the line search space. Our approach employs exclusion regions around critical user clusters in congested areas, iterates a sequential congestion-based grouping algorithm, and applies a group-member-swapping procedure. It provides max-min fairness for ground users. Extensive numerical studies have shown that our user grouping algorithm produces near-optimum beam-hopping schedules with low outage probability. It achieves an improvement of up to 13dB in the worst-case signal-to-interference and noise ratio, and doubles the zero-outage data rate, compared to benchmark approaches.*

Keywords – Inter-beam interference, max-min fairness, multibeam satellites, non-uniform user distribution, phased-array antennas

1. INTRODUCTION

Large phased-array antennas in next-generation satellites enable the formation of multiple dynamic spot beams over spatial and time dimensions [1, 2, 3]. In contrast to current systems with static beam patterns formed by feed-horn antennas [4], the phased-array antennas offer the ability to serve non-uniformly distributed users with high-capacity spot beams on demand in an adaptive manner. These capabilities are needed for supporting future broadband applications over wide rural and remote geographical areas [5, 6].

In this paper, we address the problem of coordinating the adaptive spot beams. In rural and remote areas, ground users are non-homogeneously distributed across the landscape, typically clustered together in small neighborhoods. Ideally, if there are enough spot beams, each beam should be pointed directly toward a user cluster. However, the number of spot beams per satellite is limited by the number of Radio Frequency (RF) amplifier chains feeding the phased array, and as such, not all clusters can be served simultaneously. To serve all users, the spot beams must either be broadened to provide a bigger footprint on the ground leading to a waste of the satellite resources, or time-shared via beam hopping. Beam coordination is crucial for next-generation multibeam satellites that fully utilize and reuse the available spectrum among the satellite's active beams. The challenge is to assign and point the satellite's beams to mitigate

Inter-Beam Interference (IBI) and provide max-min fairness for ground users.

The majority of existing multibeam coordination schemes use the inherited approach of selecting the beam directions from a fixed grid, in a cellular structure. The methods in [7, 8, 9, 10, 11, 12] primarily revolved around beam hopping, and took no account of the spatial heterogeneity in the network. Large spatial isolation distances were employed between beams using the same frequency band (up to four times the beam radius or more). The approach does not utilize the dynamic beam steering capability of phased-array antennas and cannot allocate network resources to adapt to dynamically changing traffic demands. The methods in [13, 14, 15] used frequency segmentation to divide the available spectrum into multiple frequency chunks. Adjacent beams were assigned to different frequencies. The approach leads to a waste of frequency resources with spatially heterogeneous traffic loads since the frequency bands allocated to sparsely populated regions are usually underutilized. The methods in [16, 17] used symbol precoding, where IBI was zeroed in the digital domain. The efficiency of precoding depended on achieving a sufficiently large separation between the beams to ensure orthogonal channels.

More recent beam coordination schemes have taken a cell-free design approach. In [18], a joint user grouping and beam design algorithm was proposed to optimize the direction and beamwidth of a multibeam system; however, IBI was not considered as the approach was limited to a single active beam per time slot. In [19], a multi-

beam system was proposed for non-uniformly distributed ground users, based on the classical four-color frequency reuse framework. The proposed scheme added some flexibility to the fixed-cell approach by dividing the beam locations in highly dense user locations into smaller beams in size without changing the predesigned fixed-cell beam locations. In [20], a modified-cell approach was used to shorten the packet queuing delay by forming user clusters and directing the spot beams to the cluster centroids. The approach aimed to deliver uniform data rates; however, it was constrained by enforcing a strict requirement of having a large spatial isolation distance between the active spot beams. In [21] a so called p-centers problem algorithm was used to minimize the number of beam positions subject to a predefined requirement on the radius of the beam position; however, it also imposed a large spatial isolation distance requirement.

In this paper, we adopt an optimized cell-free design approach and focus on solving the user-cluster grouping problem to achieve max-min fairness in service delivery for multibeam-hopping satellite systems. At a fundamental level, the grouping problem is related to a set partitioning optimization problem, that is combinatorial and Non-deterministic Polynomial-time-hard (NP-hard) [22, 23, 24]. Well-known approximation algorithms for the set-partitioning problem include random and local search [25, 26, 27] and clustering algorithms [28, 29, 30]. These do not directly translate to our user-cluster grouping problem, due to the constraints on the number of RF chains on the satellite payload and the objective of achieving max-min fairness.

We develop a line search procedure to identify near-optimum groupings for heterogeneous traffic demands. We start by proposing a congestion-based sequential grouping algorithm that groups user clusters based on a threshold distance (for exclusion region radius), starting from the areas with the highest density of users. The threshold is then updated, and the process is iterated to effectively sample the feasible user-cluster grouping solution space. To guide the choice of threshold, we provide a necessary condition to determine the boundaries of the line search space for the threshold distance. Finally, the result is augmented by a low-complexity group-member-swapping procedure.

Our algorithm integrates seamlessly with the Geostationary (GEO) satellite physical layer for analog and hybrid beamforming, as well as with other resource allocation algorithms. It only requires location information to identify near-optimum user-cluster groupings. The proposed approach generates user-cluster groupings with well-separated user clusters with minimal IBI and provides max-min fairness for ground users.

Extensive numerical studies have shown that our user grouping algorithm produces near-optimum beam-hopping schedules with low outage probability. It achieves an improvement of up to 13 dB in the worst-case Signal-to-Interference and Noise Ratio (SINR), and doubles the zero-outage data rate, compared to benchmark approaches. In addition, it achieves up to 84.4% reduction in the offered SINR variance compared to benchmark grouping algorithms.

2. SYSTEM MODEL

This section introduces the user distribution model, satellite antenna structure, the optimum user-cluster grouping problem, and the downlink channel model for serving distributed ground users using beam hopping.

2.1 User distribution and satellite antenna model

Consider the downlink of a high throughput GEO satellite system equipped with a Uniform Phased Array (UPA) antenna. The satellite provides broadband services for geographically-dispersed ground users using a limited number of RF chains, denoted by K. The ground users in remote and rural areas are typically located in clusters, where each cluster corresponds to a concentrated area spanning tens to a few hundred square kilometers, such as townships scattered across a remote area. Each cluster must be served by a beam, and the users in the cluster access the service using Time Division Multiple Access (TDMA).

Consider N geographically-dispersed ground user clusters in $\mathcal{S}$, over a large area on the Earth's surface, denoted as $\mathcal{S} \subset \mathbb{R}^2$. We represent the user clusters by the set $\mathcal{U} = \{u_1, \dots, u_N\}$, which is a finite subset of $\mathcal{S}$. The traffic distribution is heterogeneous, with some areas in $\mathcal{S}$ densely populated by user clusters while others have fewer user clusters.

To provide a service to user clusters, the satellite is equipped with a square UPA antenna which consists of $M \times M$ antenna elements as shown in Fig. 1, where

Table 1 – List of notations used in the paper.

Notation	Meaning
$\mathcal{S}$	Geographical area considered, $\mathcal{S} \subseteq \mathbb{R}^2$
$\mathcal{U}$	Set of user clusters in $\mathcal{S}$
N	Number of user clusters in $\mathcal{U}$
K	Number of the satellite's RF chains
S	Number of user-cluster groups
u_n	nth user cluster, $u_n \in \mathcal{U}, n \in [1:N]$
$\mathcal{G}_i$	ith user-cluster group, $i \in [1:S]$
α_i	Beam dwell-time fraction for group $\mathcal{G}_i$
$\gamma_{k,i}$	SINR at the kth user cluster of group $\mathcal{G}_i$

the beam directions are steered by applying a vector of phase shifts to the array elements [31]. The directivity and beamwidth of the generated spot beams are directly related to M. Generating narrower beams with a higher directivity gain and a lower level of interference between the active beams requires a larger value for M and therefore a larger array size. The number of RF chains on the satellite determines the maximum number of simultaneously active beams and the number of hops required to provide service for all user clusters in $\mathcal{U}$.

2.2 User-cluster grouping optimization

Our focus is on a beam-hopping satellite network scenario, where $N \gg K$. Therefore, the satellite's RF chains must be shared across time and space amongst multiple clusters in different hops. Here, beam hopping involves changing the direction of the beams to serve different groups of user clusters in a round-robin fashion, where each group size is equal to or less than K. Because there are many more user clusters than the number of beams, the motivation is to serve the adjacent user clusters in different groups to avoid inter-beam interference resulting from adjacent or overlapping beams. The main challenge here is to efficiently group the user clusters to avoid serving adjacent clusters or illuminating overlapping beams at the same time and therefore to ensure sufficient service quality with minimal inter-beam interference.

Before we discuss the problem of grouping the user clusters in the beam-hopping setting, we first define the concept of user-cluster grouping formally in the following definition:

Definition 1. *An (S, K)-grouping of $\mathcal{U} = \{u_1, \ldots, u_N\}$ is a collection of S disjoint subsets $\mathcal{G}_1, \ldots, \mathcal{G}_S$ of $\mathcal{U}$ that satisfy $|\mathcal{G}_i| \leq K$ for all $i \in [1:S]$. We say an (S, K)-grouping $\{\mathcal{G}_1, \ldots, \mathcal{G}_S\}$ of $\mathcal{U}$ is complete if $\bigcup_{i=1}^{S} \mathcal{G}_i = \mathcal{U}$. We also say it is a min-hop grouping if $S = \lceil \frac{N}{K} \rceil$.*

Completeness is a critical property for the grouping algorithm to ensure that all users are serviced in each beam-hopping cycle. Incomplete groupings result in unserved users, which renders it impossible to construct a beam-hopping plan covering all user clusters in $\mathcal{U}$. Such incomplete groupings will be referred to as having *leftover* user clusters in the remainder of the paper. Additionally, the min-hop grouping property plays a crucial role in minimizing the number of hops needed in a beam-hopping cycle. We note that the integer value $\lceil \frac{N}{K} \rceil$ is the minimum number of groups that need to be formed to serve all users by using K RF chains.

To maximize the collective performance of the satellite network, it is crucial to effectively select user-cluster groupings that minimize the inter-beam interference amongst the concurrently active beams. To this end, our objective is to solve the following user location grouping optimization problem:

$$
\begin{aligned}
\max_{\mathcal{G}_1, \ldots, \mathcal{G}_S} \quad & \min_{\substack{u,v \in \mathcal{G}_i: \\ u \neq v, i \in [1:S]}} \quad \|u - v\|_2 \\
\text{subject to} \quad & \bigcup_{i=1}^{S} \mathcal{G}_i = \mathcal{U} \\
& \mathcal{G}_i \bigcap \mathcal{G}_j = \emptyset, \forall i \neq j \in [1:S] \\
& |\mathcal{G}_i| \leq K, \forall i \in [1:S] \\
& S = \lceil \tfrac{N}{K} \rceil
\end{aligned}
\tag{1}
$$

where the first constraint ensures that the grouping is complete and the second constraint is to confirm that every cluster is included in no more than one group. The third constraint is due to the number of RF chains on the satellite and the last constraint ensures the min-hop grouping property. An important observation is that a user grouping that attains the optimal value of *(1)* will inherently establish max-min fairness among the user clusters. Consequently, verifying a fairness condition will be an intrinsic feature of our proposed user grouping algorithm in Section 3.

The unconstrained variant of the optimization problem in *(1)* is a version of the well-known non-convex combinatorial optimization problem [24]. Solving this problem falls within the class of NP-hard problems [32]. Achieving the optimal solution necessitates an exhaustive search algorithm, which, particularly in scenarios involving sizable N and K values, demonstrates super-exponential complexity. Other approaches can include greedy, clustering-based, and random search algorithms [26]. However, these approaches do not directly apply in our case due to the constraints involved in *(1)*.

2.3 Downlink channel model

Our model of UPA consists of $M \times M$ equispaced isotropic antenna elements as shown in Fig. 1. We set the distance between adjacent antenna elements to $\frac{\lambda}{2}$, where λ is the wavelength. This particular separation between antenna elements ensures that a beam can be formed with only one main lobe in the intended direction [31]. Without loss of generality, we take the origin as the Earth center and position the satellite along the z-axis at point $s = (0, 0, L + R_e)^\top$, where $R_e = 6371$ [km] is the Earth radius and $L = 35786$ [km] is the satellite altitude.

We model the UPA antenna orientation to coincide with the xy-plane (at the satellite altitude), looking downwards from the satellite to the Earth. The antenna elevation angle $\theta \in [0, \frac{\pi}{2}]$ is defined to be the angle between the beam pointing direction $r = (r_1, r_2, r_3)^\top \in \mathbb{R}^3$, which is a unit vector, and the negative z-axis. The azimuth angle $\phi \in [0, 2\pi)$ is the angle between the projection of the beam pointing direction on the xy-plane and the positive
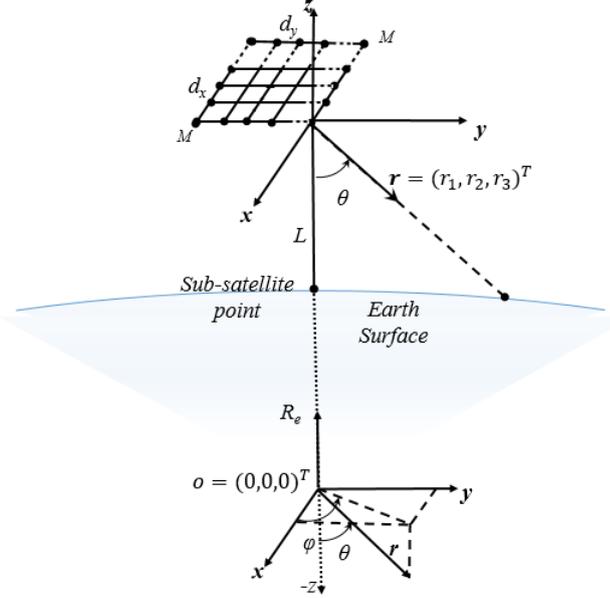
Fig. 1 – A pictorial illustration of a satellite serving ground users by using a square uniform planar array antenna.

x-axis.

Consider an (S, K)-grouping of $\mathcal{U}$, denoted by $\mathcal{G}_1, \ldots, \mathcal{G}_S$, that satisfies the constraints in *(1)*. After applying analog beamforming shifts, the normalized GEO satellite[1] downlink channel $H = \left[h_{k,l} \right]_{k,l=1}^{K}$ for user clusters in $\mathcal{G}_i = \left\{ \vec{u}_{1,i}, \ldots, \vec{u}_{K,i} \right\}$ is given by [31]

$$h_{k,l} = \frac{1}{M^2} \frac{\sin\left(\frac{M \Psi_{k,l}}{2}\right)}{\sin\left(\frac{\Psi_{k,l}}{2}\right)} \frac{\sin\left(\frac{M \Omega_{k,l}}{2}\right)}{\sin\left(\frac{\Omega_{k,l}}{2}\right)}, \qquad (2)$$

where $\Psi_{k,l} = \pi \left(\sin \theta_{l,i} \cos \phi_{l,i} - \sin \theta_{k,i} \cos \phi_{k,i} \right)$, $\Omega_{k,l} = \pi \left(\sin \theta_{l,i} \sin \phi_{l,i} - \sin \theta_{k,i} \sin \phi_{k,i} \right)$ and $\left(\theta_{k,i}, \phi_{k,i} \right)$ is the elevation-azimuth angle pair from the satellite to the user cluster $u_{k,i} \in \mathcal{G}_i$.

It is easy to see that $H \in \mathbb{R}^{K \times K}$ is a symmetric channel matrix. We can write the input-output relationship of the downlink satellite channel in matrix form according to

$$y = \sqrt{G} H D x + w, \qquad (3)$$

where $x = (x_1, \ldots, x_K)^\top \in \mathbb{C}^K$ is the vector of transmitted symbols, $y = (x_1, \ldots, x_K)^\top \in \mathbb{C}^K$ is the vector of received symbols (one for each user cluster), $D \in \mathbb{C}^{K \times K}$ is the digital beamforming matrix as in Fig. 2, G is the *total* downlink channel gain accounting for antenna gains at the transmitter and receiver, as well as the path loss

[1]Next-generation LEO satellites will also have multiple beams and will have a similar challenge to coordinate these beams considering the motion of LEO satellites with respect to the ground users and the different shapes of the beam footprints of LEO satellites.
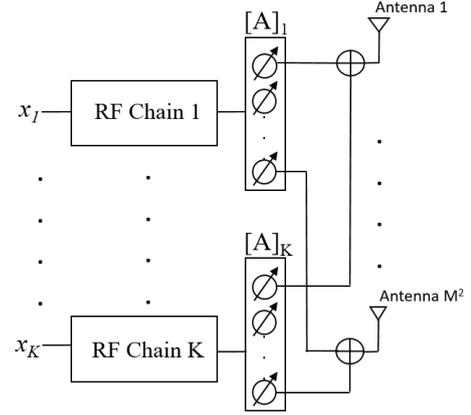
Fig. 2 – Block diagram of the satellite transmitter using a UPA antenna. The vector $a_i \in \mathbb{C}^{M^2}$ is the vector of phase shifts applied to the antenna elements to form the beam-carrying RF chain 1 data symbols.

and $w \sim \mathcal{CN}\left(0, \sigma^2 I_K\right)$ is a circularly symmetric additive Gaussian noise vector with power σ^2 at the receivers. In our numerical analysis in Section 4, we will select D to be both a zero-forcing filter and identity matrix to illustrate the hybrid and analog beamforming performance of our user location grouping algorithm.

It is worth recalling that user clusters within the same group receive services concurrently. This is the reason why we formulated the downlink input-output relationship for a specific user-cluster group in *(3)*. Using *(3)*, we can write the SINR at each user cluster in $\mathcal{G}_i$ and then calculate the data rates that can be delivered to the user clusters. It is important to note that the user clusters in group $\mathcal{G}_i$ share the same TDMA time slot, and therefore their serving beams interfere with each other. This interference should be accounted for when calculating the SINR at each user cluster $\vec{u}_{k,i}$ in $\mathcal{G}_i$.

Specifically, the SINR at user cluster $u_{k,i}$ is given according to [31]

$$\gamma_{k,i}(H) = \frac{\left| h_k^\top d_k \right|^2}{\mathsf{SNR}^{-1} + \sum_{l \neq k} \left| h_k^\top d_l \right|^2}, \qquad (4)$$

where $h_k^\top$ is the kth row of H, d_k is the kth column of D, $\mathsf{SNR} = \frac{GP}{\sigma^2}$ is the normalized signal-to-noise-ratio and P is the transmission power per RF chain. The data rate delivered to user cluster $u_{k,i}$ is then equal to

$$R_{k,i}(H) = \alpha_i W \log_2 \left(1 + \gamma_{k,i}(H) \right), \qquad (5)$$

measured in bits per second, where W is the transmission bandwidth and α_i is the fraction of time the user-cluster group $\mathcal{G}_i$ is served in each beam-hopping cycle, which we call *beam dwell-time fraction*.

3. THE PROPOSED USER-CLUSTER GROUPING ALGORITHM FOR MULTIBEAM HOPPING

In this section, we present the details of our User-Cluster Grouping (UCG) algorithm, optimizing the groups of user clusters served simultaneously in multibeam-hopping GEO satellite networks.

Our algorithm runs in polynomial time to produce a near-optimum grouping solution for the user-cluster grouping optimization problem given in *(1)*, which has super-exponential complexity for general N (number of user clusters) and K (number of RF chains). In particular, our algorithm converts the combinatorial user-cluster grouping optimization problem in *(1)* to a line search process in terms of the *exclusion region* size $\rho > 0$, a parameter of our algorithm.

3.1 Problem complexity and inefficiency of exhaustive search

We first show why the exhaustive search is inefficient in finding an optimum user-cluster grouping. We do this by establishing a fundamental scaling law for the size of the set of feasible groupings, which we denote by Σ, to search over. We note that Σ consists of all possible user-cluster groupings in the form $G = \{\mathcal{G}_1, ..., \mathcal{G}_S\}$ that satisfy the constraints in *(1)*.

Theorem 1. *For any given $K \geq 1$ (fixed), the number of feasible user groupings in Σ, which we denote by $|\Sigma|$, grows according to*

$$\lim_{N \to \infty} \frac{\log(|\Sigma|)}{S \log S} = K - 1, \qquad (6)$$

where $N, S \in \mathbb{N}$ satisfy the relation $S = \lceil \frac{N}{K} \rceil$.

Proof. See Appendix 5. □

The scaling law given in Theorem 1 reveals that for $K \geq 2$ (the key practical scenario we focus on in this paper), the number of feasible groupings within Σ grows super-exponentially as the number of user clusters in the network grows. In particular, for any $\epsilon > 0$, it shows that $|\Sigma| \geq \exp((K - 1 - \epsilon) S \log S)$ when N is large enough. Hence, the exhaustive search has super-exponential complexity to search over Σ to find the optimum user-cluster grouping $G^\star$ that solves *(1)*. This poses a significant bottleneck to the use of exhaustive search for user-cluster grouping in multibeam-hopping GEO satellite networks that consist of widely-dispersed large numbers of user clusters.

3.2 Congestion-Based Sequential Grouping (CB-SG) algorithm

Our approach to resolving the computational complexity problem established in the previous section is to iterate a congestion-based sequential grouping algorithm that we propose to sample the feasible user-cluster groupings Σ. The CB-SG algorithm is a key building block of our solution, the UCG algorithm, to group ground user clusters. We present the CB-SG algorithm in Algorithm 1, along with its main properties outlined below.

Operation of the CB-SG algorithm depends on a congestion measure $\Omega : \mathcal{U} \mapsto \mathbb{R}_+$ to prioritize the user clusters from crowded and highly dense areas in the grouping process. We select Ω to be

$$\Omega(u; \mathcal{U}) = \sum_{v \in \mathcal{U} \setminus \{u\}} \frac{1}{\|u - v\|_2^2} 1_{\|u-v\|_2 \leq r}. \qquad (7)$$

$\Omega(u; \mathcal{U})$ can be interpreted as a weighted sum of neighboring clusters of u that lie in a disc of radius $r > 0$ and centered at u, with weights inversely proportional to the distances. The parameter r determines the neighborhood radius to compute the congestion measure, which we take to be equal to the beam diameter. As illustrated in Algorithm 1, the CB-SG algorithm ranks the user clusters based on their congestion measure and assigns those with higher congestion level $\Omega(u; \mathcal{U})$ to the groups earlier in the grouping process (lines 6 and 7 in Algorithm 1).

Another important element of our CB-SG algorithm is the exclusion regions. The exclusion region of a cluster of users describes the spatial isolation of this cluster, where no other user cluster can be gathered with it in the same group [33]. For each $u \in \mathcal{U}$, it is defined as a disc of a radius ρ that is centered at u.

Let $\Pi(u, \rho) \triangleq \{v \in \mathcal{U} : \|u - v\|_2 \leq \rho\}$, which is the set of user clusters within the exclusion region of $u \in \mathcal{U}$. The CB-SG algorithm guarantees that user clusters within each other's exclusion regions are assigned to different groups. This is achieved by excluding $\Pi(u, \rho)$ from the pool of available user clusters that can be grouped with u, a step performed after u is allocated to a user-cluster grouping (line 8 of Algorithm 1).

The following theorem establishes the key properties of the CB-SG algorithm that we build upon to develop our final user-cluster grouping algorithm in the next section.

Theorem 2. *The following holds for the CB-SG algorithm:*

- *The CB-SG algorithm is a sequential grouping algorithm in the sense that if $G = \{\mathcal{G}_1, ..., \mathcal{G}_S\}$ is a set of groupings generated by the CB-SG algorithm, they satisfy $\mathcal{G}_i \subset \mathcal{U} \setminus \bigcup_{j=1}^{i-1} \mathcal{G}_i$ for all $i \in [2 : S]$.*

- *For ρ sufficiently small, the CB-SG algorithm generates*

Algorithm 1 CB-SGA: Congestion-based sequential grouping of user clusters

Input: $\mathcal{U}, K, \rho$

1: **Initialize:** $S = \lceil \frac{|\mathcal{U}|}{K} \rceil$, $\mathcal{G}_0 = \emptyset, \ldots, \mathcal{G}_S = \emptyset$, leftover = **False**
2: **for** $i = 1 : S - 1$ **do**
3:      $\mathcal{U}_{\text{rest}} \leftarrow \mathcal{U} \setminus \sum_{j=0}^{i-1} \mathcal{G}_j$
4:      $\mathcal{U}_{\text{pool}} \leftarrow \mathcal{U}_{\text{rest}}$
5:      **while** $|\mathcal{G}_i| \neq K$ & $\mathcal{U}_{\text{pool}} \neq \emptyset$ **do**
6:          select $u \in \arg\max_{v \in \mathcal{U}_{\text{pool}}} \Omega(v; \mathcal{U}_{\text{rest}})$
7:          $\mathcal{G}_i \leftarrow \mathcal{G}_i \bigcup \{u\}$
8:          $\mathcal{U}_{\text{pool}} \leftarrow \mathcal{U}_{\text{pool}} \setminus \Pi(u, \rho)$
9:      **end while**
10: **end for**
11: $\mathcal{U}_{\text{rest}} \leftarrow \mathcal{U} \setminus \sum_{i=0}^{S-1} \mathcal{G}_i$
12: **if** $|\mathcal{U}_{\text{rest}}| \leq K$ **then**
13:      $\mathcal{G}_S \leftarrow \mathcal{U}_{\text{rest}}$
14: **else**
15:      leftover = **True**
16: **end if**
17: **Return:** $G = \{\mathcal{G}_1, \ldots, \mathcal{G}_S\}$, leftover

---

a feasible user-cluster grouping $G = \{\mathcal{G}_1, \ldots, \mathcal{G}_S\} \in \Sigma$.

- *For a given $\rho > 0$, the CB-SG algorithm returns leftover as true if there exists $u \in \mathcal{U}$ such that $\left| \Pi\left(u, \frac{\rho}{2}\right) \right| \geq S + K$.*

Proof. See Appendix 5. $\qquad\square$

3.3   User-cluster grouping algorithm

Having established the key properties for the CB-SG algorithm, we now focus on our complete algorithm, which we call the UCG algorithm, that iterates the CB-SG algorithm over a range of exclusion region radii to sample the feasible groupings Σ using a line search procedure. Our algorithm is illustrated in Algorithm 2.

Creation of the solution search space: We start the discussion with how we set an upper boundary for the exclusion region radius in the UCG algorithm to sample Σ. As ρ becomes large, the CB-SG algorithm produces user-cluster groups that have well-separated user clusters in each group. The main bottleneck of using large ρ as an input to the CB-SG algorithm is that the pool of available user clusters that can be grouped is exhausted quickly when ρ is large. This leads to leftover user clusters that cannot be put in a group in the final hop.

The third property of the CB-SG algorithm established in Theorem 2 gives us a necessary condition on how large ρ can be chosen. This property indicates that whenever there exists a user cluster $u \in \mathcal{U}$ with more than $S + K - 1$ other user clusters in a disc of radius $\frac{\rho}{2}$ centered at u, the CB-SG algorithm cannot generate an (S, K)-grouping of $\mathcal{U}$ without any leftover user clusters. Utilizing this property,

Algorithm 2 UCGA: User-cluster grouping algorithm

Input: $\mathcal{U}, K$

1: **Initialize:** ρ_+ set according to *(8)*, $\rho_- = D_B$, $\tilde{\Sigma} = \emptyset$ and Fairness = **False**
2: **for** $\rho = \rho_+ : -\Delta\rho : \rho_-$ **do**
3:      $[G, \text{leftover}]$=CB-SGA($\mathcal{U}, K, \rho$)
4:      **if** leftover = **False then**
5:          $\tilde{\Sigma} \leftarrow \tilde{\Sigma} \bigcup \{G\}$
6:          **if** $\mathcal{F}(G) = 1$ **then**
7:              Fairness=**True**
8:              **Break**
9:          **end if**
10:      **end if**
11: **end for**
12: $\tilde{G}$ = UC-SELECT($\tilde{\Sigma}$)
13: $G^\star$ = UC-SWAP($\tilde{G}$)
14: **Return:** $G^\star = \{\mathcal{G}_1^\star, \ldots, \mathcal{G}_S^\star\}$
15: **procedure** UC-SELECT($\tilde{\Sigma}$) $\triangleright$ Sub-routine UC-SELECT
16:      $\tilde{G} \in \arg\max_{G \in \tilde{\Sigma}} d_{\min}(G)$
17:      **Return:** $\tilde{G}$
18: **end procedure**
19: **procedure** UC-SWAP($\tilde{G}$) $\triangleright$ Sub-routine UC-SWAP
20:      **repeat**
21:          find $\mathcal{G}_w \in \tilde{G}$ and $u_w, v_w \in \mathcal{G}_w$ such that $\|u_w - v_w\|_2 = d_{\min}(\tilde{G})$
22:          SwapGain $= \max_{\substack{\mathcal{G} \in \tilde{G}: \\ \mathcal{G} \neq \mathcal{G}_w}} \max_{\substack{v \in \mathcal{G}, \\ u \in \{u_w, v_w\}}} d_{\min}(\{\mathcal{G}_w \cup \{v\} \setminus \{u\}, \mathcal{G} \cup \{u\} \setminus \{v\}\})$ $- d_{\min}(\tilde{G})$
23:          **if** SwapGain> 0 **then**
24:              update $\tilde{G}$ by executing the best swap achieving SwapGain
25:          **end if**
26:      **until** convergence or max iteration count
27:      $G^\star \leftarrow \tilde{G}$
28:      **Return:** $G^\star$
29: **end procedure**

---

we select the upper exclusion radius according to

$$\rho_+ = \sup\left\{\rho > 0 : \left|\Pi\left(u, \frac{\rho}{2}\right)\right| < S + K, \forall u \in \mathcal{U}\right\}. \quad (8)$$

We set the lower exclusion radius ρ_- to D_B, where D_B is the spot beam diameter. Our simulations indicate that D_B is an efficient choice to resolve nearby user clusters and generate a feasible grouping in Σ. In practical scenarios, a value of $\rho < D_B$ leads to overlapping active beams, and hence our algorithm does not attempt to sample Σ to produce a grouping solution for $\rho < D_B$.

Having ρ_+ and ρ_- initialized as described above, the proposed UCG algorithm invokes the CB-SG algorithm iteratively to generate a solution search space $\tilde{\Sigma} \subset \Sigma$ by scanning the range of exclusion region radii $[\rho_-, \rho_+]$ with step-

size $\Delta\rho$ in descending order. These steps are illustrated in lines 2-5 in Algorithm 2. The step-size $\Delta\rho$ for updating the exclusion region radius can be selected to balance the solution quality and the algorithm run-time. For a practical GEO satellite network scenario, our simulations indicate that 1 km as a step size for $\Delta\rho$ is a good choice.

Early stopping: Our design of the UCG algorithm also allows early termination of the CB-SG algorithm iteration stage if a fairness stopping criterion is met, as illustrated in lines 6-7 in Algorithm 2. To this end, our fairness metric $\mathcal{F}$ is a binary variable that is given by

$$\mathcal{F}(G) = 1_{\frac{d_{\max}(G)-d_{\min}(G)}{d_{\max}(G)} \leq \epsilon_0}, \qquad (9)$$

where $G = \{\mathcal{G}_1, \dots, \mathcal{G}_S\} \in \Sigma$ is a feasible user-cluster grouping, ϵ_0 is the largest acceptable variation in the minimum mutual distance across user-cluster groups, and $d_{\max}(G)$ and $d_{\min}(G)$ are defined according to

$$d_{\max}(G) \triangleq \max_{\mathcal{G} \in G} \min_{u,v \in \mathcal{G}: u \neq v} \|u-v\|_2, \qquad (10)$$

$$d_{\min}(G) \triangleq \min_{\mathcal{G} \in G} \min_{u,v \in \mathcal{G}: u \neq v} \|u-v\|_2. \qquad (11)$$

The proposed fairness condition is designed to achieve a certain degree of max-min fairness among the user-cluster groups. This is accomplished by attaining a predefined level of regularization below ϵ_0 in the minimum mutual distance variation between the worst and best user-cluster groups. This early termination step can be skipped by selecting $\epsilon_0 < 0$ at the expense of an increase in the algorithm run-time.

Grouping selection and user-cluster swapping: Having established the solution search space $\tilde{\Sigma}$, our algorithm operates on $\tilde{\Sigma}$ to produce the final user-cluster grouping solution $G^\star = \{\mathcal{G}_1^\star, \dots, \mathcal{G}_S^\star\}$ by calling UC-SELECT and UC-SWAP sub-routines (lines 12 and 13 in Algorithm 2). The sub-routine UC-SELECT returns the user-cluster grouping that achieves the highest $d_{\min}(G)$ over $\tilde{\Sigma}$.

The primary objective of the sub-routine UC-SWAP is to have fine improvements for the best user-cluster grouping $\tilde{G}$ in $\tilde{\Sigma}$ by means of a local search around $\tilde{G}$. This is achieved by swapping one of the least favorable user clusters u_w and v_w that give rise to $\|u_w - v_w\| = d_{\min}(\tilde{G})$ with a user cluster in another group in $\tilde{G}$ that results in the maximum improvement for $d_{\min}(\tilde{G})$. By our design, UC-SWAP improves $d_{\min}(\tilde{G})$ monotonically in each iteration and converges to the final user-cluster grouping $G^\star$ quickly (around 6-7 iterations in our simulations).

Complexity analysis: We conclude this section by providing a computational complexity analysis for our algorithm in Algorithm 2. This result is formally stated in the following theorem.

Theorem 3. *The complexity of our UCG algorithm given in Algorithm 2 grows according to $\mathcal{O}(N^3)$ with the number of user clusters in a GEO satellite network.*

Proof. See Appendix 5. $\qquad\square$

Remark 1: The computational complexity of the UCG algorithm can be reduced to $\mathcal{O}(N^2)$ by performing a one-time computation of congestion measures for all user clusters within $\mathcal{U}$ and subsequently using the same measures without any further updates during the grouping process in the CB-SG algorithm.

Remark 2: We note that the beam coordination problem does not need to be solved in real time. The proposed UCG algorithm reruns whenever there is a change in the geographical distribution of the user clusters on the ground to update the user-cluster groupings that can be served simultaneously. These location changes occur at slow timescales.

4. NUMERICAL RESULTS

4.1 Simulation setup

We consider a multibeam GEO satellite equipped with a 252×252 UPA antenna designed to serve non-uniformly distributed user clusters spanning a geographical region of 4000 km $\times$ 2000 km. This choice of the UPA antenna leads to a D_B (beam diameter) of around 250 km, with an array dimension of 1.68 m $\times$ 1.68 m on the satellite payload (at operating frequency 20 GHz with half-wavelength spacing between antenna elements). We recall that D_B is a key parameter of our UCG algorithm given in Algorithm 2 to generate the solution search space. The other system-level simulation parameters that we use throughout this section are provided in Table 2.

In Fig. 3, we present an example user-cluster distribution scenario based on the population data for regional,

Table 2 – Simulation settings

Parameter	Value
Geographical area (Km^2)	4000 Km $\times$ 2000 Km
Location	(25° S, 135° W)
Satellite location	140° W (GEO)
Frequency band (GHz)	20 (Ka-band)
UPA antenna size	252×252
Transmission bandwidth (MHz)	500
Satellite beam power (watts)	20
Transmit antenna gain(dB)	52
Receive antenna gain(dB)	42
Number of RF chains	16
Satellite beam diameter (km)	250
Noise Temperature (K)	290
Number of user clusters	256

rural, and remote Australia [34]. We obtain the average performance figures for our UCG algorithm by means of Monte-Carlo simulations based on many such realizations of a non-uniform ground user-cluster distribution. Specif-

(a)

(b)
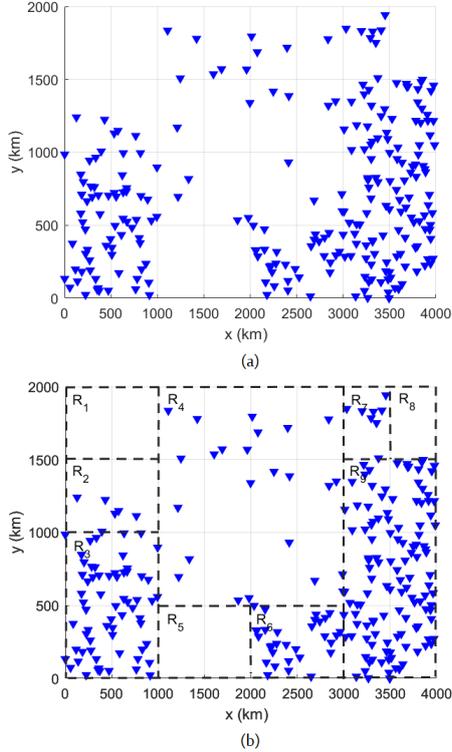
Fig. 3 – Non-uniform user-cluster distribution. (a) An example realization of 256 user clusters that are non-uniformly distributed over a square network area $\mathcal{S}$ with dimensions 4000 km $\times$ 2000 km. (b) The user-cluster populations in different subregions of $\mathcal{S}$, where R_i is the ith subregion and $\lambda_i = 0, 12, 64, 8, 0, 64, 32, 0, 81.33$ is the user-cluster density for R_i, $i \in [1 : 9]$, (measured in terms of number of user clusters per km^2).

ically, to generate randomly distributed user-cluster locations with a non-uniform distribution, we partition the overall network area into nine geographical subregions. We then distribute user clusters across these subregions uniformly at random with varying user-cluster densities from one region to another, as depicted in Fig. 3(b). The density of user clusters in each subregion is computed based on the population data provided in [34].

For detailed performance comparison and analysis, we evaluate the quality of the grouping solution produced by our UCG algorithm against the following benchmark algorithms:

- Exhaustive search (for illustrating the UCG algorithm's near-optimum performance)

- Max-Min Distance Greedy (MMDG) algorithm

- Iterative K-means (IKM) algorithm

- Low-Complexity Co-Channel Interference-Free Beam-Hopping Design (LCCF-BHD) [21]

- Fixed Cell Beam-Hopping Design (FC-BHD)

We illustrate the MMDG algorithm in Algorithm 3, which greedily optimizes the minimum distance to the existing user clusters in the group $\mathcal{G}_i$, $i \in [1 : S - 1]$ when selecting the subsequent user cluster to include in this group. The IKM algorithm iterates the K-means algorithm [28] S times to form a group of K user clusters during each iteration. This is achieved by selecting K user clusters that are nearest to the centroids generated by the K-means algorithm, assigning them to the same group, and subsequently removing them from the pool of available user clusters for the next iteration.

The LCCF-BHD algorithm is implemented as in algorithm 4 in [21] with uniform traffic across all user clusters. The FC-BHD is implemented as shown in Fig. 4, where each color represents a set of cells that can be serviced simultaneously by up to 16 active beams. In Fig. 4 there are 170 cells, each representing a beam with a 250 km diameter. The cell centers align with the beam centers, covering the non-zero-density regions of the 4000 km x 2000 km network area. We will employ the LCCF-BHD and FC-BHD for the outage performance comparison with our UCG algorithm.

To ascertain the near-optimum grouping performance of the UCG algorithm, we choose $N = 12$ and $K = 4$ due to the super-exponential computational complexity of the exhaustive search, as we have established in Theorem 1. For all other performance comparisons, we choose $N = 256$ and $K = 16$.

The performance measures that we use throughout this section are the minimum mutual separation distance between user clusters within the same group (and thus served simultaneously), user-cluster SINRs (across the entire population), SINRs of the worst users and the outage probability. In addition to evaluating the overall performance, we study the performance specifically for the worst user clusters to illustrate the fair scheduling advan-
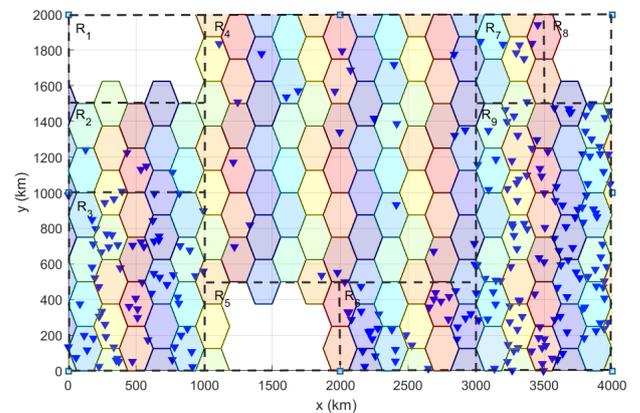
Fig. 4 – Coverage of the fixed-cell beam-hopping design

Algorithm 3 MMDGA: Max-min distance greedy grouping algorithm for user-cluster grouping.

Input: $\mathcal{U}, K$

1: **Initialize:** $S = \lceil \frac{|\mathcal{U}|}{K} \rceil, \mathcal{G}_1 = \emptyset, \dots, \mathcal{G}_S = \emptyset, \mathcal{U}_{\text{pool}} = \mathcal{U}$
2: **for** $i = 1 : S - 1$ **do**
3:      select $u \in \arg\max_{v \in \mathcal{U}_{\text{pool}}} \Omega\left(v; \mathcal{U}_{\text{pool}}\right)$
4:      $\mathcal{G}_i \leftarrow \{u\}$
5:      $\mathcal{U}_{\text{pool}} \leftarrow \mathcal{U}_{\text{pool}} \setminus \{u\}$
6:      **while** $|\mathcal{G}_i| \neq K$ **do**
7:          select $u \in \arg\max_{w \in \mathcal{U}_{\text{pool}}} \min_{v \in \mathcal{G}_i} \|w - v\|_2$
8:          $\mathcal{G}_i \leftarrow \mathcal{G}_i \bigcup \{u\}$
9:          $\mathcal{U}_{\text{pool}} \leftarrow \mathcal{U}_{\text{pool}} \setminus \{u\}$
10:     **end while**
11: **end for**
12: $\mathcal{G}_S \leftarrow \mathcal{U}_{\text{pool}}$
13: **Return:** $G = \{\mathcal{G}_1, \dots, \mathcal{G}_S\}$

---

tage of our UCG algorithm.

4.2 The minimum mutual separation distance performance

First, we evaluate the performance of our UCG algorithm in terms of the minimum mutual separation distance for multiple random realizations of the user-cluster locations from the non-uniform user-cluster distribution as described above. The performance of our grouping algorithm is compared with the exhaustive search, the MMDG algorithm, and the IKM algorithm.

4.2.1 The near-optimality of the UCG algorithm solution

We compare our proposed UCG algorithm with the *optimum* exhaustive search solution to confirm its near-optimal performance in Fig. 5. This comparison is held only for a small set of user clusters due to the super-
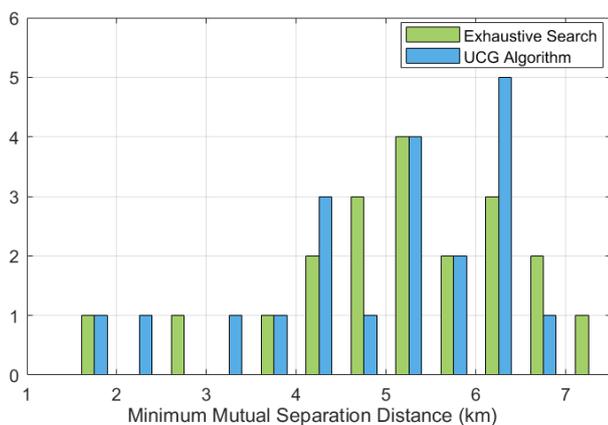
Fig. 5 – Histograms of the minimum mutual distance separation for the exhaustive search and the proposed UCG algorithm for $N = 12$, $K = 4$, and 20 realizations of the uniformly distributed user-cluster locations over a geographical area of 20 km × 20 km.

Table 3 – Statistical comparison between the minimum mutual distance of the exhaustive search and the proposed UCG algorithm

Statistical Parameter	Exhaustive Search	UCG algorithm
25$^{\text{th}}$ Percentile	4.3248	4.2129
Median	5.2811	5.1848
75$^{\text{th}}$ Percentile	6.2116	6.0457
95$^{\text{th}}$ Percentile	7.0298	6.6342
Mean	5.1635	4.9348
Variance	1.9608	2.0061

exponential complexity of the exhaustive search. The considered data set is limited to 20 different realizations of 12 user clusters uniformly distributed over a geographical area of 20 km × 20 km and 4 RF chains.

Fig. 5 shows that the minimum mutual distance separation between the group members in the optimal solution obtained by the exhaustive search ranges from 2 km to 7 km, with the majority of occurrences around 5 km. The mean and median values for the minimum mutual distance achieved by the exhaustive search are 5.1635 and 5.2811, respectively. The proposed UCG algorithm can achieve minimum mutual distances in the range of 1.5 km to 6.5 km with the most repeated occurrences around 6 km. The mean and median values for the minimum mutual distance achieved by our UCG algorithm are 4.9348 and 5.1848, respectively. These results indicate the potential of our UCG algorithm to perform close to the optimum solution generated by the exhaustive search. Table 3 provides the summary statistics for the minimum mutual distance achieved by the exhaustive search and the proposed UCG algorithm, further solidifying the near-optimum performance of our UCG algorithm.

4.2.2 Performance comparison with the MMDG algorithm and IKM algorithm

Next, we evaluate the minimum mutual distance performance of our proposed UCG algorithm in comparison to that of the MMDG algorithm and IKM algorithm for larger networks. Fig. 6 provides the histograms for the minimum mutual distances that can be achieved by the user-cluster groupings of the proposed UCG algorithm, the MMDG algorithm, and the IKM algorithm. This figure is generated for 100 realizations of 256 user-cluster locations non-uniformly distributed over the 4000 km × 2000 km network area. There are 16 RF chains on the satellite to serve the user clusters.

Fig. 6 clearly shows that our UCG algorithm achieves superior performance when compared to the MMDG algorithm and IKM algorithm in terms of the minimum mutual separation distance. Specifically, this figure illustrates that the proposed UCG algorithm consistently maintains a minimum mutual distance ranging between 210 km and 350 km, and exceeds the beam diameter $D_B = 250$ km for

97% of the time. This observation implies that our UCG algorithm achieves user-cluster scheduling that can avoid the occurrence of overlapping beams successfully with a very high probability.

On the other hand, the MMDG algorithm can achieve a minimum mutual distance separation of only up to 70 km and the IKM algorithm of up to 200 km. The presence of small values for the minimum mutual distance in the MMDG algorithm and IKM algorithm implies the existence of disadvantaged user clusters served simultaneously without adequate separation for effective interference mitigation. This observation further suggests that the beam-hopping plan derived from these grouping algorithms will not be adequate in catering to the demands of all user clusters.

The main reason for the low minimum mutual distance performance for the MMDG algorithm and the IKM algorithm is that they cannot maintain fairness between the groups formed at different stages of the grouping process. To illustrate this point further, we present a visual comparison of the colored mapping of the grouping solutions produced by our UCG algorithm and the MMDG algorithm for a particular realization of user-cluster locations in Fig. 7. Each color represents a group of user clusters. The color code for the first and last groups is as follows: The dark blue color represents the first formed group, while the dark crimson color represents the last group of user clusters.

The figure shows that the MMDG algorithm gives the advantage of sizeable mutual separation distances for the first-formed groups over the last groups. The user clusters coded by the dark blue color are nicely separated for the MMDG algorithm. This comes at the cost of the diminished mutual distance of the user clusters in the last groups, where the user clusters are very close to each

other. This unfair scheduling of the user clusters results in a degradation of the level of the offered service to the user clusters in the last groups. On the other hand, thanks to its max-min fairness advantage, this scenario is avoided by the proposed UCG algorithm. The user clusters are fairly distributed over the formed groups. Even for the last group represented by the dark crimson color, the user clusters are well separated.

Fig. 7 – A particular realization of grouping solutions produced by the UCG algorithm and the MMDG algorithm ($K = 16$). (a) Non-uniformly distributed 256 user-cluster locations over the network area 4000 km $\times$ 2000 km. (b) The grouping solution produced by the UCG algorithm. (c) The grouping solution produced by the MMDG algorithm.

Fig. 6 – Histograms of the minimum mutual distance separation for the UCG algorithm, the MMDG algorithm, and the IKM algorithm for $N = 256$, $K = 16$, and 100 realizations of the non-uniformly distributed user-cluster locations over a geographical area of 4000 km $\times$ 2000 km.

4.2.3 The effect of the number of RF chains on the minimum mutual distance performance

We study the effect of the number of RF chains on the minimum mutual distances achieved by our UCG algorithm, the MMDG algorithm, and the IKM algorithm in Fig. 8. The performance curves in this figure are obtained by averaging over 100 realizations of non-uniformly distributed 256 user-cluster locations over the network area 4000 km × 2000 km.

The figure shows that the UCG algorithm achieves a higher value of the minimum mutual distance than those achieved by the MMDG algorithm and IKM algorithm for all values of the RF chains. In particular, the UCG algorithm can achieve minimum mutual distance up to 705 km and 452 km higher than those achieved by the MMDG algorithm and the IKM algorithm (when $K = 4$), respectively.

As the number of RF chains increases, the minimum mutual distances achieved by all three algorithms decrease. This is due to the larger group sizes formed by the algorithms when K is bigger, which increases the possibility of serving nearby user clusters with smaller mutual distances. It is important to note that our UCG algorithm consistently achieves an average minimum mutual distance exceeding the 250 km beam diameter for up to $K = 24$. On the other hand, the MMDG algorithm and IKM algorithm struggle to sustain a minimum mutual distance separation above 250 km, even for a small number of RF chains.

Overall, it is clear from Fig. 8 that both the MMDG algorithm and IKM algorithm have a higher possibility of grouping nearby user clusters for simultaneous service provisioning compared to our proposed UCG algorithm. Among the three algorithms, the MMDG algorithm exhibits the poorest minimum distance separation performance due to its greedy nature and unfair scheduling of the user clusters in the final groups.
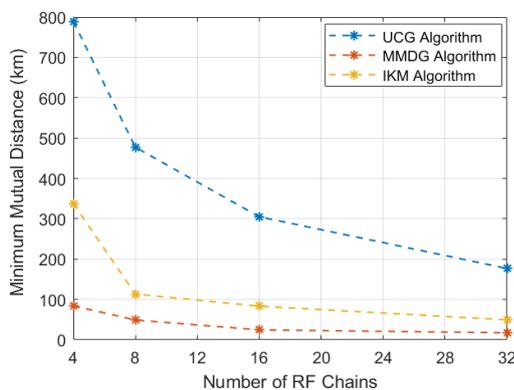
4.3 SINR and outage performance

In this section, we provide a thorough investigation of the SINR and outage performance of our proposed UCG algorithm, relative to other benchmark algorithms. The user-cluster SINRs are an important performance measure for the grouping algorithms that directly relate to multibeam-hopping GEO satellite throughput. After S groups $\mathcal{G}_1, \ldots, \mathcal{G}_S$ of user clusters are formed, we compute the SINR, denoted by $\gamma_{k,i}(H)$, at user cluster $u_{k,i} \in \mathcal{G}_i$ according to *(4)*, where H is the channel matrix given by *(2)*. We select the digital beamforming matrix D to be both a zero-forcing filter (hybrid beamforming) and identity matrix (pure analog beamforming) to compute $\gamma_{k,i}(H)$ in our numerical results below.

The outage curves are plotted by converting user-cluster SINRs to data rates using Shannon's formula according to *(5)* and comparing the data rates with the target rates.

4.3.1 SINR *histograms*

We start the discussion by studying the histograms of the offered SINR values by the UCG, MMDG, and IKM algorithms, which are given in Fig. 9. They are generated by using analog beamforming, without any interference cancellation in the digital domain, for the entire user-cluster population, as well as for the worst ten percent of the user clusters and the worst user cluster. We study the SINR distribution for the disadvantaged user clusters to highlight the fairness advantage of our UCG algorithm in meeting user demands compared to the other algorithms.

The figure illustrates that the SINR values offered by our proposed UCG algorithm range from 6 dB to 16 dB. In contrast, the SINR values provided by the MMDG and IKM algorithms exhibit a wide variation, spanning from -7 dB to 16 dB and from -4 dB to 16 dB, respectively. In particular, the SINR variance across the entire population of user clusters provided by our UCG algorithm is 84.4% and 60.9% less than the SINR variance provided by the MMDG and IKM algorithms, respectively. Our UCG algorithm reduces the SINR variance significantly without sacrificing the mean SINR performance.

In addition to less variation in the offered SINR, our UCG algorithm also improves the SINR provided for the worst user cluster, as indicated in Fig. 9. In particular, the mean value for the worst user-cluster SINR achieved by our UCG algorithm is 8.3066 dB. On the other hand, it is equal to -4.1390 dB and 0.0863 dB for the MMDG and IKM algorithms, respectively. The median values for the worst user-cluster SINRs for the UCG, MMDG, and IKM

Fig. 8 – The minimum mutual distance between the user clusters for the proposed UCG algorithm, the MMDG algorithm, and the IKM algorithm for different numbers of RF chains ($N = 256$). The curves are obtained by averaging over 100 realizations of non-uniformly distributed user-cluster locations over a geographical area of 4000 km × 2000 km.
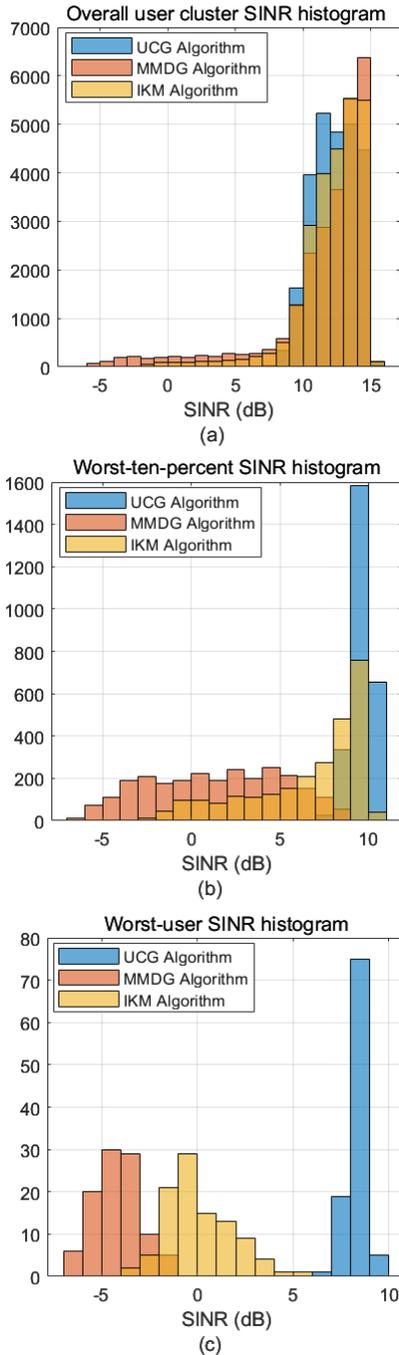
Fig. 9 – Histograms of the offered SINRs for the UCG algorithm, the MMDG algorithm, and the IKM algorithm for $N = 256$, $K = 16$, and 100 realizations of the non-uniformly distributed user-cluster locations over a geographical area of 4000 km $\times 2000$ km. (a) SINR histograms for the entire user-cluster population. (b) SINR histograms for the worst ten percent of the user-cluster population.(c) SINR histograms for the worst user cluster.

algorithms are 8.3121 dB, -4.0963 dB, and -0.2777 dB, respectively.

The superiority of our UCG algorithm is not exclusively limited to the worst user-cluster SINR value. The offered SINR to the worst ten percent of the user clusters by our UCG algorithm is in the range of 7 dB to 11 dB, while it

Table 4 – Statistical comparison between the offered SINR by the proposed UCG algorithm, MMDG algorithm, and IKM algorithm to the worst ten percent of the user-cluster population

Statistical Parameter	UCG alg.	MMDG alg.	IKM alg.
25th Percentile	9.2854	−1.6589	4.7717
Median	9.6622	1.5680	7.9492
75th Percentile	10.0013	4.5097	9.1900
95th Percentile	10.3311	7.2431	9.8127
Mean	9.5898	1.4393	6.6364
Variance	0.3028	13.8717	10.4921

ranges from -7 dB to 9 dB, and from -3 dB to 11 dB for the MMDG and IKM algorithms, respectively. To corroborate this observation further, we provide key summary statistics for the SINR values provided to the bottom ten percent of the user clusters by the three algorithms in Table 4. Our UCG algorithm significantly outperforms the MMDG and IKM algorithms across all statistical measures given in Table 4.

4.3.2 Effect of the number of RF chains on the SINR performance

Next, we focus on the effect of the number of RF chains on the SINR performance for the UCG, MMDG, and IKM algorithms. We study the SINR values offered only to the worst ten percent of the user-cluster population (see Fig. 10). For the entire population, the mean SINR performance for the three algorithms is similar. For the worst user cluster, our UCG algorithm outperforms the others more significantly.

In Fig. 10(a), we plot the average SINR performance for the UCG, MMDG, and IKM algorithms when analog beamforming is used to steer the beams towards the served user-cluster locations. As shown in this figure, our UCG algorithm attains superior SINR values for all values of K, surpassing those of the MMDG and IKM algorithms by up to 8.16 dB and 2.94 dB, respectively.

In Fig. 10(b), we plot the average SINR performance for the UCG, MMDG, and IKM algorithms when zero-forcing hybrid beamforming is employed for interference cancellation in the digital domain, along with beam steering in the direction of the served user clusters using analog beamforming. Similar to the analog beamforming scenario, our UCG algorithm outperforms the MMDG and IKM algorithms significantly in the hybrid beamforming case.

More importantly, we observe that our UCG algorithm derives the greatest advantage from the use of hybrid beamforming when compared to the MMDG and IKM algorithms. This is because the interference cancellation ability of the zero-forcing hybrid beamforming in the digital domain is limited to the scenarios in which the si-
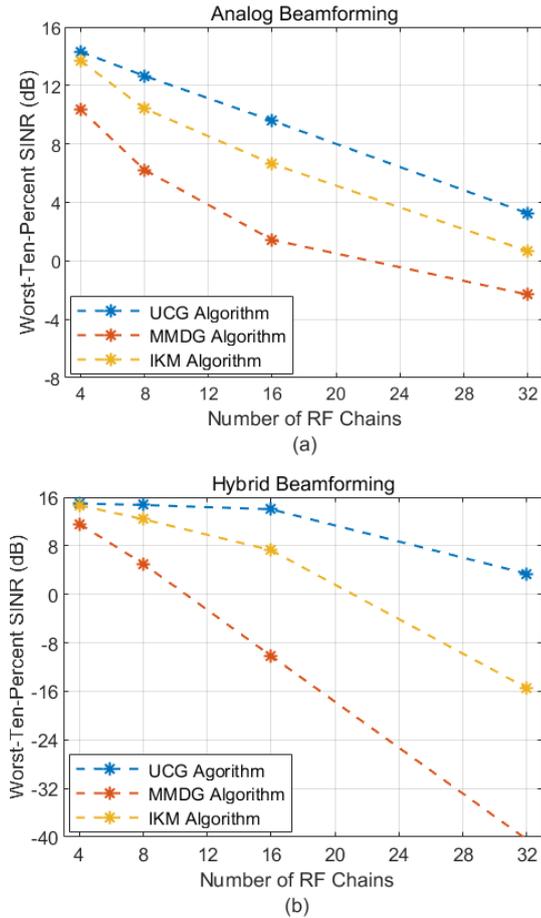
Fig. 10 – The worst ten percent SINR for the proposed UCG algorithm, the MMDG algorithm, and the IKM algorithm as a function of the number of RF chains for $N = 256$ and 100 realizations of the non-uniformly distributed user-cluster locations over a geographical area of 4000 km $\times$ 2000 km. (a) Average analog beamforming SINR. (b) Average hybrid beamforming SINR.

multaneously active beams are sufficiently spaced apart. It is also clear that the SINR does not change with k for the small number of RF chains (up to $k = 16$) while it decreases for the case of ($k = 32$). This is because of the bigger size of the user-cluster group and as the active beams approach each other, as is the case with user clusters in the same group that are close in proximity, the zero-forcing hybrid beamforming struggles to cancel the IBI without significantly compromising the direct channel gain.

Our UCG algorithm maintains high values for the minimum mutual distance as a function of K (see Fig. 8), enabling effective interference cancellation with zero-forcing hybrid beamforming. This observation underscores the critical role of our UCG algorithm in the context of hybrid beamforming for multibeam-hopping GEO satellite networks. On the other hand, the minimum mutual distance gets smaller and the MMDG and IKM algorithms cannot maintain sufficient separation among the user clusters in the same group when the number of

RF chains on the satellite payload increases (see Fig. 8 again). Hence, they are unsuited for multibeam-hopping GEO satellite networks with zero-forcing hybrid beamforming.

4.3.3 Outage performance

Finally, we investigate the outage performance of our UCG algorithm, comparing it to the MMDG and IKM algorithms, as well as the FC-BHD and LCCF-BHD schemes. We define the average outage percentage, which is the main performance metric that we focus on in this part, according to

$$F(R_\tau) \triangleq \frac{1}{N} \mathsf{E} \left[\sum_{i=1}^{S} \sum_{k=1}^{|\mathcal{G}_i|} \mathbb{1}_{R_{k,i}(H) < R_\tau} \right], \qquad (12)$$

where R_τ is the target (requested) rate, $R_{k,i}(H)$ is the achieved data rate at the kth user cluster belonging to the ith group, which is computed according to *(5)*, and the expectation is taken over random user-cluster locations.

The beam dwell-time fractions (i.e., α_i's in *(5)*) for the UCG, MMDG, and IKM algorithms are set to $\frac{1}{16}$ for all the user-cluster groups. For the LCCF-BHD, they are determined based on the beam-hopping cycle for each particular realization of user-cluster locations, uniformly across all user-cluster groups. The FC-BHD scheme's cells with no user cluster receive zero beam dwell-time fractions. The time fraction allocations for the remaining cells are determined based on the beam-hopping cycle for each particular realization of user-cluster locations, again uniformly across all the cells.

We illustrate the average outage percentage of our UCG algorithm as a function of the target rate, in comparison to MMDG and IKM algorithms and LCCF-BHD and FC-BHD schemes, in Fig. 11. The outage percentage curves are averaged over 100 random realizations of 256 non-uniformly distributed user-cluster locations. Among the three algorithms and two beam-hopping cell designs, the FC-BHD scheme is the only fixed-cell design approach for multibeam hopping.

Starting with Fig. 11(a) where we show the outage performance with analog beamforming, we observe that our UCG algorithm outperforms the other approaches. In particular, our UCG algorithm achieves zero outage percentage up to 100 Mbps target rate, with a sharp transition to a hundred percent outage afterward. This observation indicates that our UCG algorithm delivers a uniform rate to non-uniformly distributed user clusters by achieving max-min fairness in separation distance across different user-cluster groups. We call the data rate at which a sharp transition to full outage occurs *zero-outage* data rate.

The MMDG and IKM algorithms, however, both lead to outage before reaching the target rate of 100 Mbps, with a
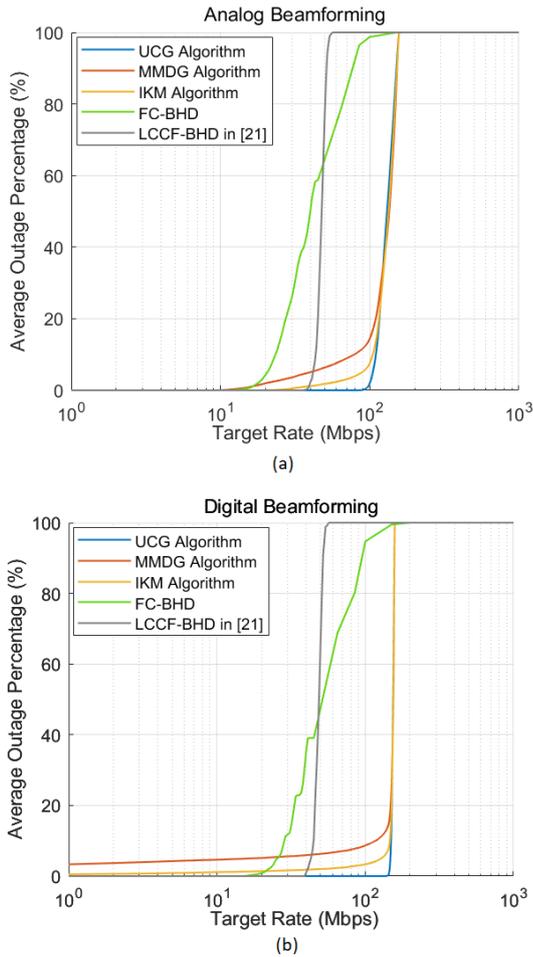
Fig. 11 – User-cluster outage percentage for the UCG, MMDG, and IKM algorithms, as well as the LCCF-BHD and FC-BHD schemes for 100 realizations of 256 non-uniformly distributed user-cluster locations over a geographical area of 4000 km × 2000 km. (a) Analog beamforming outage performance. (b) Hybrid beamforming outage performance.

sharp transition to a hundred percent outage at this target rate. This is because they cannot regularize the minimum mutual separation distance to the same extent as our UCG algorithm does (see Fig. 6), which inevitably causes outages for a certain percentage of the user-cluster population at almost all target rates above 10 Mbps.

The LCCF-BHD scheme has a sharp transition in the outage percentage at a target rate of 50 Mbps, half the target rate achieved by our UCG algorithm. This observation suggests that the LCCF-BHD scheme is capable of providing a uniform rate to non-uniformly distributed user clusters, similar to our UCG algorithm. However, its outage performance is considerably more limited in comparison, as it reaches full outage at a rate half that of our UCG algorithm. The reason for the LCCF-BHD scheme's limited performance is the large spatial isolation separation it employs, which restricts the optimal utilization of all available RF resources on the satellite payload. As a result, the beam-hopping cycle for the LCCF-BHD scheme is extended, leading to reduced data rates.

The FC-BHD scheme exhibits the poorest outage performance among the others. It has a ramp-type transition from zero outage to the full hundred percent outage with a median target rate of 50 Mbps. In particular, the data rates delivered by the FC-BHD scheme are very heterogeneous across the user clusters. This is due to the rigid fixed-cell design paradigm adopted in the FC-BHD scheme, which hinders the equitable distribution of communication resources across regions with varying population densities. Last but not least, we note that the FC-BHD scheme is also affected adversely by the existence of cell-edge user clusters since the user clusters can be located anywhere inside a cell. In contrast, our UCG algorithm, along with other cell-free designs, does not suffer from the cell-edge user-clusters phenomenon since the beams are always centered at the user-cluster locations.

In Fig. 11(b), we present the corresponding outage percentage curves obtained with zero-forcing hybrid beamforming. As illustrated by the figure, the same qualitative trends continue to hold for the hybrid beamforming scenario, reflecting those observed with analog beamforming. Specifically, our UCG algorithm outperforms the other methods, achieving delivery of regulated data rates and displaying a higher target rate with a sharp transition to full outage. The benefit of interference cancellation in the digital domain by hybrid beamforming is that the zero-outage data rate rises to 200 Mbps for our UCG algorithm, as well as for the MMDG and IKM algorithms. In contrast, hybrid beamforming does not offer advantages to the FC-BHD and LCCF-BHD schemes as they inherently enforce substantial spatial separation between active beams. These schemes aim to cancel interference in the analog domain by design, rendering them less responsive to hybrid beamforming for multibeam-hopping satellite networks.

4.4 The computational complexity

Table 5 shows the numerical results of the computational complexity of the proposed algorithm compared to the exhaustive search and the benchmark algorithms for different user-cluster set sizes and a fixed number of RF chains ($K = 16$). It is clear from the table that the exhaustive search has a super-exponential computational complexity to coordinate the multiple beams and this complexity is significantly reduced when our UCG algorithm is used. Moreover, although the proposed algorithm has a higher computational complexity than, or the same as, the MMDG algorithm or the IKM algorithm, respectively, it has the advantage of superior performance as shown in figures 6, 8, 9, 10 and 11.

Table 5 – Numerical results of the computational complexity of the exhaustive search, the proposed UCG algorithm, the MMDG and IKM algorithms for $K = 16$

N	Exh. Search	UCG	MMDG	IKM
Order	$\exp\left((K-1)\,S\log S\right)$	N^3	$N^2\log N$	N^3
128	$8.0131643e+46$	$100,352$	$34,528$	$100,352$
256	$3.244182e+125$	$802,816$	$157,826$	$802,816$
512	∞	$6,422,528$	$710,218$	$6,422,528$

5. CONCLUSION

In this paper, we have developed a novel user-cluster grouping algorithm for multibeam-hopping GEO satellite networks with non-uniformly distributed user-cluster locations. Our method utilizes a density-based sequential grouping algorithm, effectively transforming the exponentially-complex optimum grouping problem into a line search procedure. Operating within polynomial time, our algorithm generates near-optimum user-cluster groupings, ensuring well-separated user-cluster locations for efficient simultaneous service provisioning.

One of the key performance metrics that we have focused on in the paper is the minimum mutual distance separation, serving as a proxy for evaluating service quality delivered to the worst-case user-cluster locations. We have shown that our algorithm produces a distribution for the minimum mutual distance metric that is statistically similar to what is generated by the exhaustive search method. In particular, the mean and median values for the minimum mutual distance separation obtained by our algorithm and the exhaustive search are 4.9348 and 5.1635 (mean), and 5.1848 and 5.2811 (median), respectively. In terms of distribution variance and percentile values, our algorithm and the exhaustive search method also yield similar summary statistics.

We have also evaluated the performance of our user-cluster grouping algorithm in terms of the offered user-cluster SINRs and data rates, in comparison to four other benchmark approaches. We have demonstrated that our algorithm surpasses the benchmark algorithms both in terms of the offered SINR values and data rates, with an improvement of up to 13 dB in SINR values provided to the worst user locations. In a beam-hopping setting, our algorithm generates a beam-hopping plan to deliver uniform rates to non-uniformly distributed user clusters, with no outage up to the target data rates 100 Mbps (pure analog beamforming) and 200 Mbps (zero-forcing hybrid beamforming).

For future work, it is suggested to consider our user-cluster grouping algorithm with LEO satellites. Moreover, modeling the communication channel as a time-varying channel is another potential future research direction.

APPENDICES

Proof of Theorem 1

For the sake of simplicity, we will assume $N = S \cdot K$. The general case where K does not divide N evenly without a remainder follows from the same arguments.

The number of all feasible groupings in Σ is given by

$$
\begin{aligned}
|\Sigma| &= \frac{1}{S!}\prod_{i=0}^{S-1}\binom{N-iK}{K} \\
&= \frac{1}{S!}\frac{N!}{(K!)^S}.
\end{aligned}
$$

Using Stirling's approximation, $|\Sigma|$ can be approximated as

$$
\begin{aligned}
|\Sigma| &\sim \frac{1}{\sqrt{2\pi S}\left(\frac{S}{e}\right)^S}\cdot\frac{\sqrt{2\pi N}\left(\frac{N}{e}\right)^N}{\left(\sqrt{2\pi K}\right)^S\left(\frac{K}{e}\right)^N} \\
&= \left(\frac{e}{\sqrt{2\pi}}\right)^S\frac{S^{(K-1)S}}{K^{\frac{S-1}{2}}}, \qquad (13)
\end{aligned}
$$

where $f(n) \sim g(n)$ for any two functions f, g means $\lim_{n\to\infty}\frac{f(n)}{g(n)} = 1$. Taking the logarithm of both sides in *(13)*, we have

$$
\begin{aligned}
\log|\Sigma| &\sim S\log\left(\frac{e}{\sqrt{2\pi}}\right) \qquad (14) \\
&+ S(K-1)\log S - \frac{S-1}{2}\log K.
\end{aligned}
$$

Dividing both sides of *(15)* by $S\log S$ and taking the limit as $N \to \infty$, we get

$$
\lim_{N\to\infty}\frac{\log|\Sigma|}{S\log S} = K - 1,
$$

which concludes the proof.

Proof of Theorem 2

The CB-SG algorithm satisfies the sequential grouping property because it removes the user-cluster groupings $\mathcal{G}_1, \dots, \mathcal{G}_{i-1}$ it constructs from $\mathcal{U}$ before forming the ith user-cluster group (line 3 in Algorithm 1).

For the second property, the first constraint in *(1)* amounts to the CB-SG algorithm returning false for the leftover variable, which always holds for $\rho < \min_{u,v\in\mathcal{U}:u\neq v}\|u - v\|$. The second constraint in *(1)* is satisfied by the sequential grouping nature of the CB-SG algorithm. The third and fourth constraints in *(1)* are trivially satisfied by the construction of the algorithm.

For the third property, assume that $\left|\Pi\left(u, \frac{\rho}{2}\right)\right| \geq S + K$ for some $u \in \mathcal{U}$. Since the CB-SG algorithm does not allow any two user clusters to be in the same group $\mathcal{G}_i$ for $i \in$

$[1 : S - 1]$ if their separation is less than ρ, there can be at most one user cluster that belongs to both $\mathcal{G}_i$ and $\Pi\left(u, \frac{\rho}{2}\right)$ for $i = [1 : S - 1]$. This implies

$$
\begin{aligned}
\left|\bigcup_{i=1}^{S-1} \Pi\left(u, \frac{\rho}{2}\right) \bigcap \mathcal{G}_i\right| &= \sum_{i=1}^{S-1} \left|\Pi\left(u, \frac{\rho}{2}\right) \bigcap \mathcal{G}_i\right| \\
&\leq S - 1, \quad\quad (15)
\end{aligned}
$$

where the first equality holds because $\mathcal{G}_i$'s are disjoint and the inequality follows from the above observation that $\left|\mathcal{G}_i \bigcap \Pi\left(u, \frac{\rho}{2}\right)\right| \leq 1$.

To conclude the proof, we will show $\left|\mathcal{U} \bigcup_{i=1}^{S-1} \mathcal{G}_i\right| > K$, which implies that the CB-SG algorithm returns leftover as true and it cannot group all the remaining user clusters in the same group $\mathcal{G}_S$ at the final hop. To this end, we have

$$
\begin{aligned}
\left|\mathcal{U} \bigcup_{i=1}^{S-1} \mathcal{G}_i\right| &\geq \left|\Pi\left(u, \frac{\rho}{2}\right) \bigcup_{i=1}^{S-1} \mathcal{G}_i\right| \\
&= \left|\Pi\left(u, \frac{\rho}{2}\right)\right| - \left|\bigcup_{i=1}^{S-1} \Pi\left(u, \frac{\rho}{2}\right) \bigcap \mathcal{G}_i\right| \\
&\geq S + K - (S - 1) \\
&= K + 1,
\end{aligned}
$$

where the last inequality follows from *(15)* and the assumption that $\left|\Pi\left(u, \frac{\rho}{2}\right)\right| \geq S + K$.

Proof of Theorem 3

The main computational complexity of the UCG algorithm stems from the complexity of computing the congestion measure $\Omega\left(u; \mathcal{U}_{\text{rest}}\right)$ in the CB-SG algorithm for the user clusters that remain to be grouped. The complexity of computing $\Omega\left(u; \mathcal{U}_{\text{rest}}\right)$ for a single user cluster is $\mathcal{O}\left(N\right)$. This computation is performed for each of the $\mathcal{O}\left(N\right)$ user clusters in each step and it is repeated for $S = \lceil \frac{N}{K} \rceil$ times, which leads to complexity $\mathcal{O}\left(N^3\right)$.

The computational complexity of the UC-SELECT procedure is only $\mathcal{O}\left(N\right)$ since the group sizes are bounded by K and there are $S = \lceil \frac{N}{K} \rceil$ groups for each $G \in \tilde{\Sigma}$. Similarly, the computational complexity of the UC-SWAP procedure is also $\mathcal{O}\left(N\right)$.

ACKNOWLEDGEMENT

REFERENCES

[1] H. Fenech, S. Amos, and T. Waterfield. "The role of array antennas in commercial telecommunication satellites". In: *Proc. 10th European Conference on Antennas and Propagation (EuCAP)*. 2016, pp. 1–4.

[2] J. Warshowsky, C. Kulisan, and D. Vail. "20 GHz phased array antenna for GEO satellite communications". In: *MILCOM 2000 Proceedings. 21st Century Military Communications. Architectures and Technologies for Information Superiority (Cat. No.00CH37155)*. Vol. 2. 2000, 1187–1191 vol.2.

[3] K. J. Maalouf and E. Lier. "Theoretical and experimental study of interference in multibeam active phased array transmit antenna for satellite communications". In: *IEEE Transactions on Antennas and Propagation* 52.2 (2004), pp. 587–592.

[4] S. Rao, M. Tang, and C.-C Hsu. "Multiple beam antenna technology for satellite communications payloads". In: *Applied Computational Electromagnetics Society Journal* 21 (Nov. 2006), pp. 353–363.

[5] O. Kodheli, E. Lagunas, N. Maturo, S. K. Sharma, B. Shankar, J. F. Mendoza Montoya, J. C. M. Duncan, D. Spano, S. Chatzinotas, S. Kisseleff, J. Querol, L. Lei, T. X. Vu, and G. Goussetis. "Satellite Communications in the New Space Era: A Survey and Future Challenges". In: *IEEE Communications Surveys & Tutorials* 23.1 (2021), pp. 70–109.

[6] S. Fu, J. Gao, and L. Zhao. "Collaborative Multi-Resource Allocation in Terrestrial-Satellite Network Towards 6G". In: *IEEE Transactions on Wireless Communications* 20.11 (2021), pp. 7057–7071.

[7] Y. Wang, D. Bian, J. Hu, J. Tang, and C. Wang. "A Flexible Resource Allocation Algorithm in Full Bandwidth Beam Hopping Satellite Systems". In: *Proc. 2019 IEEE 3rd Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*. 2019, pp. 920–927.

[8] A. Wang, L. Lei, E. Lagunas, S. Chatzinotas, A. I. P. Neira, and B. Ottersten. "Joint Beam-Hopping Scheduling and Power Allocation in NOMA-Assisted Satellite Systems". In: *Proc. 2021 IEEE Wireless Communications and Networking Conference (WCNC)*. 2021, pp. 1–6.

[9] Y. Han, C. Zhang, and G. Zhang. "Dynamic Beam Hopping Resource Allocation Algorithm Based on Deep Reinforcement Learning in Multi-Beam Satellite Systems". In: *Proc. 2021 3rd International Academic Exchange Conference on Science and Technology Innovation (IAECST)*. 2021, pp. 68–73.

[10] D. Wei, D. Zheng, C. Pan, and L. Yang. "Dynamic Beam Scheduling of Multibeam Low Earth Orbit Satellites Based on an Enhanced Artificial Bee Colony Algorithm". In: *IEEE Access* 10 (2022), pp. 115424–115434.

[11] L. Wang, C. Zhang, D. Qu, and G. Zhang. "Resource Allocation for Beam-hopping User Downlinks in Multi-beam Satellite System". In: *Proc. 2019 15th International Wireless Communications and Mobile Computing Conference (IWCMC)*. 2019, pp. 925–929.

[12] Z. Wu, P. Ren, and D. Xu. "Flexible Resource Allocation for Differentiated QoS Provisioning in Beam-Hopping Satellite Communications System". In: *Proc. 2022 IEEE 95th Vehicular Technology Conference: (VTC2022-Spring)*. 2022, pp. 1–5.

[13] E. S. Abass, J. V. M. Halim, and H. M. El-Hennawy. "Coded-beam Strategy for High Throughput Satellites - A Path Toward Terahertz Communications - System Model, Realization and Performance Assessment". In: *Proc. 2020 12th International Conference on Electrical Engineering (ICEENG)*. 2020, pp. 181–185.

[14] L. Lei, J. Wan, and Q. Gong. "Multibeam Antenna with Reflector for Ka/Ku Band High Throughput Satellite Applications". In: *Proc. 2021 International Conference on Microwave and Millimeter Wave Technology (ICMMT)*. 2021, pp. 1–3.

[15] L. Del Consuelo Hernandez Ruiz Gaytan, Z. Pan, J. Liu, and S. Shimamoto. "Dynamic Scheduling for High Throughput Satellites Employing Priority Code Scheme". In: *IEEE Access* 3 (2015), pp. 2044–2054.

[16] M. G. Kibria, E. Lagunas, N. Maturo, D. Spano, and S. Chatzinotas. "Precoded Cluster Hopping in Multi-Beam High Throughput Satellite Systems". In: *Proc. 2019 IEEE Global Communications Conference (GLOBECOM)*. 2019, pp. 1–6.

[17] C. Zhang, X. Zhao, and G. Zhang. "Joint precoding schemes for flexible resource allocation in high throughput satellite systems based on beam hopping". In: *China Communications* 18.9 (2021), pp. 48–61.

[18] B. Liu, C. Jiang, L. Kuang, and J. Lu. "Joint User Grouping and Beamwidth Optimization for Satellite Multicast with Phased Array Antennas". In: *Proc. GLOBECOM 2020 - 2020 IEEE Global Communications Conference*. 2020, pp. 1–6.

[19] C. Xu and Q. Du. "Resource Allocation Scheme in High Throughput Satellite Systems Based on Dynamic Beam-coverage Algorithm". In: 2022, pp. 302–307.

[20] J. Tang, D. Bian, G. Li, J. Hu, and J. Cheng. "Optimization Method of Dynamic Beam Position for LEO Beam-Hopping Satellite Communication Systems". In: *IEEE Access* 9 (2021), pp. 57578–57588.

[21] L. Lyu and C. Qi. "Beam Position and Beam Hopping Design for LEO Satellite Communications". In: *China Communications* 20 (2023), pp. 29–42.

[22] R. M. Karp. "Reducibility among Combinatorial Problems". In: *Complexity of Computer Computations: Proceedings of a symposium on the Complexity of Computer Computations, New York*. Ed. by Raymond E. Miller, James W. Thatcher, and Jean D. Bohlinger. Boston, MA: Springer US, 1972, pp. 85–103.

[23] M. R. Garey and D. S. Johnson. "Strong NP-Completeness Results: Motivation, Examples, and Implications". In: *Journal of the ACM* 25.3 (1978), pp. 499–508.

[24] J. Kleinberg and É. Tardos. *Algorithm Design*. Addison-Wesley, 2006.

[25] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms*. 2nd edition. The MIT Press, 2001.

[26] Z. B. Zabinsky. *Stochastic adaptive search for global optimization*. Boston: Kluwer Academic Publishers, 2003.

[27] H. R. Lourenço, O. C. Martin, and T. Stützle. "Iterated Local Search: Framework and Applications". In: *Handbook of Metaheuristics*. Ed. by Michel Gendreau and Jean-Yves Potvin. Springer US,Boston, MA, 2010, pp. 363–397.

[28] S. P. LLOYD. "Least Squares Quantization in PCM". In: *IEEE Transactions on Information Theory* 28.2 (1982).

[29] P. Bradley, K. Bennett, and A. Demiriz. "Constrained K-Means Clustering". In: *MSR-TR-2000-65* (Aug. 2000).

[30] M. Ester, H. P. Kriegel, J. Sander, and X. Xu. "A Density-Based Algorithm for Discovering Clusters". In: *Proc. KDD'96: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*. 1996.

[31] C. A. Balanis. *Antenna Theory: Analysis and Design*. USA: Wiley-Interscience, 2005.

[32] M. Danilova, P. E. Dvurechensky, A. V. Gasnikov, E. A. Gorbunov, S. Guminov, D. Kamzolov, and I. Shibaev. "Recent Theoretical Advances in Non-Convex Optimization". In: *ArXiv* abs/2012.06188 (2020).

[33] A. Mokhtar and M. Azizoglu. "On the Downlink Throughput of a Broadband LEO Satellite Network with Hopping Beams". In: *IEEE Communications Letters* 4 (2000).

[34] Austarlian Institute of Family Studies. "Families in regional, rural and remote Australia [Fact Sheet]". In: *Austarlian Government: Australian Institute of Family Studies Website* (2011). [accessed online: 3-August-2023].

AUTHORS

Heba Shehata is a PhD Student at Macquarie University, Australia. She received her BSc and MSc degrees in electrical engineering from Alexandria University, Egypt. She also worked as a research assistant at Qatar University, Qatar before joining Macquarie University. She received the best paper award at the 28th Asia-Pacific Conference on Communications (APCC), 2023. Her research interests include satellite communications, wireless communications, resource management, and spectrum sensing and sharing.

Hazer Inaltekin is a senior lecturer at Macquarie University. He received his B.S. degree (High Honors) in electrical and electronics engineering from Bogazici University, Istanbul, Turkey, and his M.S./Ph.D. degree in electrical and computer engineering from Cornell University, Ithaca, NY. Prior to joining Macquarie University, he held various researcher and faculty positions in Australia, Europe, and the United States. His research interests include airborne networks, satellite communications, fog/edge computing, IoT, wireless communications, and information theory.

Iain B. Collings (Fellow, IEEE) received a B.E. degree in electrical and electronic engineering from The University of Melbourne in 1992 and a Ph.D. degree in systems engineering from The Australian National University in 1995. Currently, he is a professor at the School of Engineering, Macquarie University, Sydney, Australia, where he was the Deputy Dean of Research and the Head of Department. Prior roles include the deputy chief (Research) of the Computational Informatics Division, Australian Commonwealth Scientific and Industrial Research Organization (CSIRO), and academic positions with The University of Sydney and The University of Melbourne. He has published over 340 research papers in the area of wireless digital communications. In 2009, he was awarded the Engineers Australia IREE Neville Thiele Award for Outstanding Achievements in Engineering. In 2011, he was a recipient of the IEEE CommSoc Stephen O. Rice Best Paper Award of IEEE Transactions on Communications. He served as an editor for IEEE Transactions on Wireless Communications (2002–2009), the Physical Communication Journal (Elsevier) (2008–2012), and Sensors since 2020. He has served as the co/vice-Chair of the Conference Technical Program Committees for IEEE International Conference on Communications (ICC) 2013, IEEE Vehicular Technology Conference (VTC) spring 2011, IEEE Wireless Communications and Networking Conference (WCNC) 2010, and IEEE VTC Spring 2006. He is a founding organizer of the Australian Communication Theory Workshops (2000–2023). He also served as the chair for the Joint Communications & Signal Processing Chapter in the IEEE NSW Section (2008–2010) and a secretary for the IEEE NSW Section (2010).